

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
6 November 2003 (06.11.2003)

PCT

(10) International Publication Number
WO 03/091849 A2

(51) International Patent Classification⁷: **G06F**

(21) International Application Number: PCT/US03/12832

(22) International Filing Date: 23 April 2003 (23.04.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/374,936 23 April 2002 (23.04.2002) US

(71) Applicant: **THE CLEARING HOUSE SERVICE COMPANY L.L.C.** [US/US]; 100 Broad Street, New York, NY 10004 (US).

Landview Drive, Kings Park, NY 11754 (US). **PAWEL-CZYK, Joseph, S.**; 9 Wyckoff Way, Morris Plains, NJ 07950 (US). **FRIEDMAN, Kenneth, H.**; 22 Exeter Drive, Marlboro, NJ 07746 (US). **PALMESE, Paul**; 500 Pierce Street, Oceanside NY 11572 (US).

(74) Agents: **CLAYTON, Ronald, A.** et al.; Fitzpatrick, Cella, Harper & Scinto, 30 Rockefeller Plaza, New York, NY 10112-3801 (US).

(84) Designated States (*regional*): European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR).

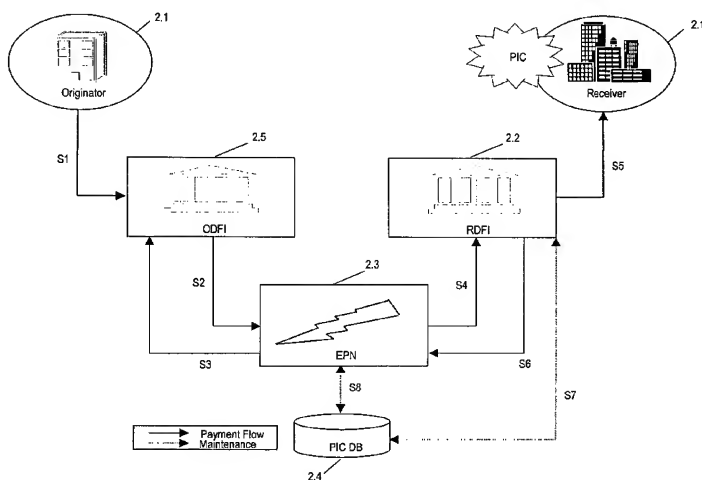
Published:

— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(72) Inventors: **THOMAS, George, F.**; 111 Middlesex Avenue, Oakdale, NY 11769 (US). **WOOD, Albert, G.**; 8

(54) Title: PAYMENT IDENTIFICATION CODE AND PAYMENT SYSTEM USING THE SAME



(57) Abstract: A method for effecting electronic payment, safeguarding banking and account information, while utilizing existing payment systems. The method comprises generating a system routing number and a payment identification code (PIC) relating to the beneficiary's account information, distributing payment identification codes to the existing payment system and financial institutions owning the account related to the payment identification codes, and the originator receiving a system routing number and the beneficiary's PIC number. The method also includes the originator communicating a payment instruction to a financial institution of the originator, wherein the payment instruction includes the system routing number beneficiary's payment identification code, the originator's financial institution receiving the payment instruction from the originator, wherein if the received PIC matches the originator's financial institution internal list of PICs, the originator's

financial institution performs an "on us" transaction, and transmitting a payment instruction to an existing payment system in a case where the received PIC does not match originator's financial institution internal list of PICs. The method also includes the existing payment system validating the received PIC, wherein if the PIC is invalid, the payment instruction is returned to the originator's financial institution, converting the PIC and system routing number to a receiving payment instruction in a case where the PIC is a valid PIC, wherein the receiving payment instruction includes the beneficiary's financial institution's routing number and the beneficiary's account number. The existing payment system transmits the receiving payment instruction to a financial institution of the beneficiary, that financial institution credits the beneficiary's account if no problem exists, and otherwise returns a receiving payment instruction to the existing payment system. Upon receipt of the returned receiving payment instruction, the existing payment system translates the receiving payment instruction into the payment instruction prior to transmitting the payment instruction back to originator's financial institution.



WO 03/091849 A2

TITLE

**PAYMENT IDENTIFICATION CODE AND
PAYMENT SYSTEM USING THE SAME**

BACKGROUND OF THE INVENTION

Field of the Invention

[0001] The present invention relates to systems and methods of making electronic payments. In particular, the present invention relates to systems and methods for making payments to and from banking institutions.

Description of the Related Art

[0002] With the emergence and adoption of the Internet and related technologies, businesses are moving toward electronic integration of supply and financial chains. Complete financial integration requires the ability to issue information-rich, secure, private and guaranteed final payments.

[0003] Consumer-to-business e-payment has grown steadily over the past years, but business-to-business (B2B) e-payment growth has been much slower. Among

the reasons for this lack of growth in B2B e-payments are the fear that e-payments would lack remittance information and other vital data, and the reluctance to give out account numbers.

[0004] The current environment for payments involving businesses and banks is primarily a paper one. The efficiency of paper processing has created a weak electronic bill payment infrastructure. Banks are at the center of the bill payment process. They hold customer accounts from which payments are authorized and are positioned to deliver the remittance information to the biller. Banks are also positioned to deliver invoice information to the biller's customers who are also the banks' customers.

[0005] In conventional non-electronic bill payment systems, where an ongoing relationship exists, a party initiating payment (hereinafter "payor") pays a debt to a biller by mailing a check in response to receipt of the biller's invoice. The term "biller" is used to refer to the "payee" or entity to be paid. Attached to most biller's invoices is a payment coupon to be returned with the check. The coupon contains at least the consumer-biller account number, as well as other information that will assist the biller, or the biller's bank, in properly crediting the consumer (i.e., the party initiating payment) with payment.

[0006] The need to improve payment systems was recognized in the late 1960s. Special committees on paperless entries were formed and alternatives to paper checks were developed. From this early work, the first automated clearing house (ACH) for the exchange of consumer-oriented paperless entries was established. The early ACH associations worked with their local Federal Reserve Bank to provide the facilities, equipment, and staff to operate an automated clearing house. Ultimately this lead to the development of the Electronic Payment Network (EPN), which is a private sector automated clearing house operator.

[0007] The ACH network is a low-cost electronic payment mechanism that can be used to pay both individual consumers and companies, regardless of size. In order to use the ACH network, bank routing information and payee (demand deposit

account (DDA) identifier) account number must be supplied. This information must either be supplied by the initiator of the payment, or must be retained by the banking system of the payor. This presents a major problem that inhibits widespread use of the ACH network because bank routing and account information of the payee is rarely conveyed to payors for use in initiating payment instructions. [0008] One type of system used in processing international and domestic payments electronically is the Clearing House Interbank Payments System (CHIPS), which was established in 1970. CHIPS is the foremost means for transferring U.S. dollars among world banks. In the CHIPS, a universal identifier (UID) number is utilized that uniquely identifies individual customer accounts. The CHIPS UID number is a six-digit number that is used to identify named accounts at depository institutions on the CHIPS.

[0009] Another system used for processing electronic payment is the Electronic Payment Network.

[0010] Because of the problems of security, authorization, authenticity, the fear that e-payments would lack remittance information and other vital data, and the reluctance of businesses to give out account numbers, there exists a need for a system and method that would enable the initiation and receipt of electronic payments with full remittance information that leverages the best features of existing electronic payment systems, such as ACH EPN and CHIPS, as a backbone to the system and method.

[0011] The need also exists for maintaining the confidentiality of account information and provide ease of maintenance when an account holder transfers from one depository institution to another.

[0012] Future enhancements to electronic bill payment will be integrated with existing systems, including the present invention, to form a complete supply and financial chain integration, as depicted in Figure 1.

[0013] As shown in Figure 1, the iClearing & Settlement (iC&S) allows for modular implementation wherein existing infrastructures/processes, such as CHIPS

and EPN, provide the basis for an electronic payment system. Envisioned is a new, future payment channel, XML of Rich Payment information that allows the Buyer's Bank and the Seller's Bank to communicate directly with the iC&S system for electronic payments. The Financial Services Solution provides adjudicated invoices to the Buyer's Bank and Rich Information (XML) to the Seller's Bank of buy and sell transactions between a Buyer and Seller.

SUMMARY OF THE INVENTION

[0014] The present invention has been made to solve the above-explained conventional problems, and it is an object of the invention to provide a secure electronic payment method.

It is another object of the invention to maintain the confidentiality of account information. The objects of the invention are realized by a method for effecting electronic payment between an originator's account and a beneficiary's account, safeguarding banking and account information, while utilizing existing payment systems, and to a system that operates in accordance with the method. The method comprises generating a system routing number and a payment identification code (PIC) relating to the beneficiary's account information, distributing a list of payment identification codes to the existing payment system and financial institutions owning the account related to the payment identification codes on the list, and the originator receiving a system routing number and the beneficiary's PIC number. The method also includes the originator communicating a payment instruction to a financial institution of the originator, wherein the payment instruction includes the system routing number beneficiary's payment identification code, the originator's financial institution receiving the payment instruction from the originator, wherein if the received PIC matches the originator's financial institution internal list of PICs, the originator's financial institution performs an "on us" transaction, and transmitting a payment instruction to an existing payment system in a case where the received PIC does not match

originator's financial institution internal list of PICs. The method further includes the existing payment system validating the received PIC against a PIC database, wherein if the PIC is invalid, the payment instruction is returned to the originator's financial institution, converting the PIC and system routing number to a receiving payment instruction in a case where the PIC is a valid PIC, wherein the receiving payment instruction includes the beneficiary's financial institution's routing number and the beneficiary's account number. The existing payment system transmits the receiving payment instruction to a financial institution of the beneficiary, the beneficiary's financial institution credits the beneficiary's account if no problem exists, and otherwise returns a receiving payment instruction to the existing payment system. Upon receipt of the returned receiving payment instruction by the existing payment system, the existing payment system translates the receiving payment instruction into the payment instruction prior to transmitting the payment instruction back to originator's financial institution, wherein the payment identification code is unique to each beneficiary's account.

[0015] According to another aspect of the present invention, a web-based payment method is provided for effecting electronic payment between an originator's account and a beneficiary's account utilizing existing payment systems. The method includes generating a payment identification code and a system routing number uniquely identifying account information of the beneficiary, distributing the payment identification code and the beneficiary's account information relating to the payment identification code to the existing payment systems, the beneficiary transmitting the payment identification code to the originator, and in response to a payment order from the beneficiary, the originator transmitting a payment instruction to the financial institution of the originator. The payment instruction includes the payment identification code of the beneficiary, and the amount to be paid. The method also includes the originator's financial institution processing and transmitting a payment instruction to an existing payment system to effect an electronic funds transfer of funds, the existing payment system converting a

payment identification code included in a payment instruction to the beneficiary's account information and forwarding a converted payment instruction to the beneficiary's financial institution, and the beneficiary's financial institution effecting an electronic funds transfer on the basis of the converted payment instruction by crediting the beneficiary's account. According to one embodiment of the invention the communications between the originator, the financial institution of the originator, the existing payment system, the beneficiary's financial institution, and the beneficiary is accomplished at least partly via the Internet, although in other embodiments, at least some or all of the communication may take place via any other suitable type of network of interest.

[0016] According to a further aspect of the present invention, a secure electronic payment method between a consumer and a biller facilitated through an existing payment system is provided. The method includes generating a payment identification code unique to a biller's account information, distributing the payment identification code to the biller and the existing payment system, the biller communicating the payment identification code to the consumer, and the consumer electronically transmitting a payment instruction via the consumer's financial institution to the existing payment system. The payment instruction preferably comprises information indicating at least source of the consumer's account information, a payment amount, and the biller's payment identification code. The method also includes the existing payment system validating the payment identification code of the biller, and, upon validating the payment identification code of the biller, the existing payment system converting the payment identification code of the biller included in the payment instruction into the biller's account information which includes the routing number of the biller's financial institution. Further steps include transmitting the converted payment instruction to the biller's financial institution, applying a credit to the biller's account in an amount corresponding to the payment amount included in the payment instruction,

and applying a debit to the consumer's account in an amount corresponding to the payment amount included in the payment instruction.

[0017] Further objects, features, and advantages of the present invention will become apparent from the following detailed description of the invention with reference to the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] The present invention will be more readily understood from a detailed description of the preferred embodiments taken in conjunction with the following figures.

[0019] Figure 1 is a block diagram of potential future integration of supply and financial chains;

[0020] Figure 2 is a flowchart of the present invention integrated with the EPN system;

[0021] Figure 3 is a flowchart of the present invention integrated with the CHIPS system;

[0022] Figure 4 is a flowchart of the present invention integrated with an alternate-CHIPS configuration;

[0023] Figure 5 is an example of a cash management entry window;

[0024] Figure 6 is an example of an entity-relationship diagram of the PIC database;

[0025] Figure 7 is a functional process model depicting approach to defining business requirements of the present invention;

[0026] Figures 8 and 9 are a flowchart of a create a bank profile;

[0027] Figure 10 is a flowchart of a delete bank profile process;

[0028] Figure 11 is a flowchart of a modify bank profile process;

[0029] Figure 12 is a flowchart of a change route process;

[0030] Figure 13 is a flowchart of a view PIC activity log process;

[0031] Figure 14 is a flowchart of a view bank profile activity log process;

- [0032] Figures 15A and 15B are a flowchart of a create user process;
- [0033] Figures 16A and 16B are a flowchart of a delete user process;
- [0034] Figures 17A and 17B are a flowchart of a modify user process;
- [0035] Figure 18 is a flowchart of a view user profile process;
- [0036] Figure 19 is a flowchart of a log on to the PIC system process;
- [0037] Figure 20 is a flowchart of a change password process;
- [0038] Figure 21 is a flowchart of a reset user process;
- [0039] Figures 22A and 22B are a flowchart of a transfer PIC process;
- [0040] Figure 23 is a flowchart of a create PIC process;
- [0041] Figures 24A and 24B are a flowchart of a close PIC process;
- [0042] Figures 25A and 25B are a flowchart of a reactivate PIC process;
- [0043] Figure 26 is a flowchart of a contest/release a PIC pending transfer process;
- [0044] Figures 27A and 27B are a flowchart of an approve a PIC activity process;
- [0045] Figures 28A and 28B are a flowchart of a find a PIC process;
- [0046] Figure 29 is a flowchart of a flowchart of a view user profile activity log process;
- [0047] Figures 30A and 30B are a flowchart of a modify PIC process;
- [0048] Figure 31 is a flowchart of a perform trusted third party validation process;
- [0049] Figure 32 is a flowchart of a validate bank profile process;
- [0050] Figures 33A and 33B are a flowchart of an approve bank profile modification process;
- [0051] Figure 34 is a block diagram of a PIC batch service.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0052] The system using the payment identification code of the present invention will have the advantage of encouraging the use of electronic payments between business buyers and sellers. To employ the concept, banks are issued a unique

payment identifier code (PIC) number for each business customer demand deposit account (DDA). As part of the issuance process, a trusted third party, such as a clearing house, for example the New York Clearing House, links sensitive information related to the seller and its individual DDA to the issued PIC number. Such a trusted third party will set up a system (hereinafter “the system”) that will, among other things, maintain and distribute PIC numbers for all that wish to use them. As a result, confidential account relationship information is masked to outside parties. Individual sellers communicate their PIC numbers to buyers who are then capable of originating payments using the PIC number. Upon receipt of a payment instruction containing a valid seller PIC number, a payment system, such as EPN or CHIPS, can then access the PIC database to retrieve associated account information required to execute the payment.

[0053] The use of the PIC number promises to deliver several benefits:

[0054] First, increased security. By safeguarding banking and account information, corporations minimize the potential for fraudulent account activity. This benefit is very important for Internet-based transactions where counter-parties do not know each other. Also, it is envisioned that the PIC number will be an integral component of the open-standards-based payment channel of the future.

[0055] Second, portability. Preferably, individual PIC numbers will remain with business customers regardless of changes in their banking relationships or account information such as address. As a result, corporations can communicate a single payment identification code to trading partners, a code that never changes.

[0056] Third, efficiency. Presently, 12 billion business-to-business checks are written each year. It is envisioned that the PIC number will be able to be used by businesses that do not employ electronic payments as part of their financial operations. Reducing the number of paper-based payments increases efficiency for all members of the financial chain.

[0057] Preferred embodiments of the system for implementing the present invention will now be discussed with reference to Figures 2 through 4. Figures 2

through 4, which are high-level block diagrams of the various parties involved in the implementation and use of the preferred embodiments, show the PIC Process Definition and how the PIC concept integrates with the existing EPN and CHIPS payment systems. Hardware systems suitable for use by the various parties are well known in the art, and one of ordinary skill in the art is capable of implementing such systems as needed. In practice, it is expected that the system will utilize systems and communication infrastructure that are already in operation. Also, the responsibilities of buyers, sellers and their respective financial institutions are outlined. While similar in concept, many of the details associated with processing PIC payments vary between EPN-based and CHIPS-based transactions. It should be noted while the use of the PIC number will be explained with reference to the existing EPN and CHIPS systems, this is for illustrative purposes only and the present invention is not limited to use with those systems.

[0058] Fig. 2 is a diagram illustrating the process utilizing the PIC of the present invention with the EPN system 2.3 As a pre-condition for the process steps described Fig. 2, a PIC number must be issued to the RDFI 2.2 for the receiver's DDA. The RDFI 2.2 communicates a system routing number and PIC number to the receiver 2.1.

[0059] An originator 2.6 receives a system routing number and PIC number from the receiver 2.1. The originator 2.6 enters the system routing number and the receiver's PIC number into the normal routing and account number fields in either the cash management system supplied by the ODFI 2.5 or through its accounts payable system. An example of a cash management entry interface is depicted in Figure 5.

[0060] Figure 5 shows a web-based cash manager system enabling the originator to make a payment request. The cash management entry in Figure 5 includes at least the following fields vendor name (beneficiary/receiver) 5.1, vendor identification number 5.2, amount of payment 5.3, a payment description 5.4, vendor bank identification 5.5, for which in the present embodiment the system

routing number will be used, and a vendor bank account number 5.6 for which in the present invention the PIC number will be used. Other fields can be added as necessary, depending on each financial institution's requirements.

[0061] At step S1 of the flowchart in Fig. 2, the ODFI 2.5 receives a payment request, identifies it as a system routing transaction by the routing number and checks the PIC number against an internal list of PIC numbers to determine if the receiver's 2.1 PIC number exists at the ODFI 2.5 (i.e., to check if it is an "on-us" transaction). If the PIC number indicates an on-us transaction, the ODFI 2.5 completes a book transfer of funds, and the ODFI 2.5 debits the originator's account.

[0062] At step S2, if a PIC number match does not exist, the ODFI 2.5 sends the payment instruction to the EPN system 2.3. At step S3, the EPN system 2.3 validates the PIC number against the PIC database 2.4. If invalid, the payment instruction is returned to the ODFI 2.5 with an appropriate error message. Once validated, the EPN system 2.3 replaces the system routing number and PIC number with the RDFI routing number and the receiver's DDA number (not shown), and at step S4, payment information is routed to the RDFI 2.2. At step S5, the RDFI 2.2 credits the receiver DDA.

[0063] At step S6, if there is a problem at the RDFI 2.2 receiving the payment, it is returned through the EPN system 2.3. The EPN system 2.3 recognizes returns specific to the system and translates the DDA number back into the system routing number and PIC number before returning the payment instruction to the ODFI 2.5.

[0064] At step S7, the RDFI 2.2 maintains PIC numbers related to their business customer accounts through one of the system's service channels. At step S8, the PIC database (not shown) on the EPN 2.3 is updated daily with changes from the master PIC database 2.4. The master PIC database 2.4 is described later.

[0065] Fig. 3 is a diagram illustrating the process utilizing the PIC of the present invention with the CHIPS system 3.4. As a pre-condition for the processing in Fig. 3, the system issues a PIC number for the beneficiary's DDA to the beneficiary's

bank 3.2. The beneficiary's bank 3.2 communicates iC&S routing number and PIC number to the beneficiary 3.1.

[0066] The originator 3.6 receives the system routing number and PIC number from the beneficiary 3.1. The originator 3.6 enters the system routing number and the beneficiary's PIC number into the normal routing and account number fields in either the cash management system supplied by originator's bank 3.3 or the accounts payable system.

[0067] At step S10, the originator's bank 3.3 (also known as the CHIPS send participant) receives a payment request and checks the PIC number against internal list of PIC numbers to determine if the beneficiary's PIC number is a business customer of the originator's bank 3.3.

[0068] At step S11, if a PIC number match does not exist, the originator's bank 3.3 sends the payment instruction to CHIPS 3.4. At step S12, CHIPS 3.4 validates the PIC number against the PIC database on CHIPS. If invalid, the payment is rejected and sent back to the originator's bank 3.3 with the appropriate error message.

[0069] At step S13, once validated, CHIPS 3.4 replaces the system routing number and PIC number with the beneficiary bank's and (in this case) the CHIPS receive participant's routing number and the beneficiary's DDA number. At step S13, all payment information is routed to the CHIPS receive participant (beneficiary bank) 3.2, and at step S14, the beneficiary bank 3.2 credits the beneficiary's 3.1 DDA.

[0070] At step S13, the beneficiary bank 3.2 maintains PIC numbers related to their customer accounts through one of the channels provided by the system (which is described later) and at step S16, the PIC database on CHIPS (not shown) is updated daily with changes from the master PIC database 3.5.

[0071] Fig. 4 is a diagram illustrating an alternative process utilizing the PIC of the present invention with the CHIPS 4.5 system (CHIPS with corresponding bank). As a pre-condition for the process depicted in Fig. 4, the system issues a PIC number to the beneficiary's bank 4.2 for the beneficiary's 4.1 DDA. The

beneficiary bank 4.2 communicates the system routing number and PIC number to receiver.

[0072] The originator 4.7 receives the system routing number and the PIC number from the beneficiary 4.1. The originator 4.7 enters the system routing number and the beneficiary's PIC number into the normal routing and account number fields in either the cash management system supplied by the originator's bank 4.3 or the accounts payable system.

[0073] At step S17, the originator's bank 4.3 and (in this case) the CHIPS send participant receives the payment request and checks the PIC number against an internal list of PIC numbers to determine if the beneficiary's 4.1 PIC number is a business customer of the originator's bank 4.3. At step S18, if a PIC number match does not exist, the originator's bank 4.3 sends the payment instruction to CHIPS 4.5.

[0074] At step S19, CHIPS 4.5 validates the PIC number against the PIC database on CHIPS. If invalid, the payment is rejected and sent back to originator's bank 4.3 with the appropriate error message. Once validated, CHIPS 4.5 replaces the routing number and PIC number with the CHIPS receive participant's routing number and the beneficiary's 4.1 DDA number.

[0075] If the beneficiary's bank 4.2 is the CHIPS receive participant, the payment is sent directly to the beneficiary bank 4.2 and the business customer DDA is credited. In the illustrated case, the beneficiary's bank 4.2 is not a CHIPS receive participant, so CHIPS 7.5 looks to the beneficiary's 4.1 PIC for predetermined CHIPS receive participant routing information.

[0076] At step S20, information on the entire CHIPS chain is included in the payment instruction (beneficiary's bank 4.2 routing number and beneficiary's 4.1 DDA number, name, address, etc.) and routed to the CHIPS receive participant 4.4. At step S21, the CHIPS receive participant 4.4 receives the payment and forwards it to the beneficiary's bank 4.2, and at step S22, the beneficiary's bank 4.2 credits the beneficiary's 4.1 DDA.

[0077] At step S23, the beneficiary bank 4.2 maintains PIC numbers related to their customer accounts through one of the channels provided by the system, and at step S24, the PIC database on CHIPS (not shown) is updated daily with changes from the master PIC database 4.6.

[0078] To use the PIC enhancement of the present invention, participant banks must fulfill certain requirements. These requirements include:

[0079] Connectivity. Initially, banks must either have an operational connection to SWIFTNet or ConnectDirect.

[0080] SWIFTNet is S.W.I.F.T's™ advanced IP-based messaging solution. SWIFTNet allows a financial institution to do business in an environment that combines the security of a private network and the guarantees of a trusted third party with the flexibility of Internet technologies.

[0081] ConnectDirect is a peer-to-peer file-based integration software for high-volume, assured file transfers. ConnectDirect automates the secure, reliable transfer of large volumes of data within and between enterprises. ConnectDirect is available only for those banks that are currently set up for file transfer. These connections are necessary to provide a secure channel for enrollment and maintenance of PIC numbers.

[0082] ACH System Enhancement (Minor). Because the PIC number hides receiving bank information, originating banks do not know when they are originating "on us" transactions. Banks wishing to filter out "on us" PIC transactions may want to make modifications to their ACH origination methods/systems. However, modifications are not required.

[0083] Resource Commitment. Banks must have the resources to supply an enrollment file extract, test the PIC system, and train their staff to service PICs.

[0084] Other System Requirements. Programming changes may be required on the sending and the receiving sides of a transaction. However, the required changes would be minor. For example, a sending bank may want to change the

cash management field labeled “account number” to “PIC number.” Or, the receiving bank may want to “track” all transactions involving the system.

[0085] For those banks that choose to pilot the PIC service, there are several additional requirements. These requirements include:

[0086] Fees. If a pilot bank cannot immediately identify and divert “on us” PIC transactions, the trusted third party can issue a credit for the estimated amount of “on-us” charges incurred during pilot activities.

[0087] Customer Identification. The pilot bank would preferably be required to identify specific business customers whom they intend to partner with during the PIC pilot for EPN. The criteria for pilot customers (buyers and sellers) include EPN/CHIPS registration. Both the buyer and seller are required to be EPN registered bank customers. This requirement prevents the situation where a buyer sends a PIC-based transaction to their bank who, in turn, send it to the Federal Reserve who will likely not be able to process the transaction during the pilot phase.

[0088] The PIC database is housed in a relational database application. Figure 6 depicts an example of a structure of the PIC database. As is appreciated, there are countless configurations for a relational database depicting a PIC database. Figure 6 only illustrates one such possible configuration, and in no way should be construed as limiting the application of the inventive systems and methods to that configuration. Figure 6 shows a PIC database with 16 tables. Table 1 is a summary of the PIC database and describes each of the tables represented in Figure 6. Figure 6 and Table 1 are illustrative only of a PIC database.

Table 1
Database Summary

| Target DBMS: | Microsoft SQL Server | | | |
|-------------------------|----------------------|---------|--------------|-------|
| Number of tables: | 16 | | | |
| Number of views: | 0 | | | |
| Number of columns: | 165 | | | |
| Number of Indexes: | 4 | | | |
| Number of foreign keys: | 20 | | | |
| last map date: | 1/2/1970 | | | |
| Extended attributes: | | | | |
| Filegroup | F1 | | | |
| Filegroup | F2 | | | |
| Filegroup | PRIMARY | | | |
| Tables | Columns | Indexes | Foreign Keys | Notes |
| PICSTATUS | 2 | 0 | 0 | |
| PICS | 1 | 0 | 0 | |
| PICACTIVITY | 41 | 2 | 7 | |
| PICACTION | 2 | 0 | 0 | |
| PICACCOUNTSTATUS | 2 | 0 | 0 | |
| PIC | 31 | 1 | 3 | |
| SYSTEMPARA | 2 | 0 | 0 | |

| | | | |
|----------------|----|---|---|
| SOURCE | 2 | 0 | 0 |
| BANKSTATUS | 2 | 0 | 0 |
| BANKROUTING | 2 | 0 | 1 |
| BANKPROFILE | 29 | 0 | 3 |
| BANKACTIVITY | 41 | 1 | 6 |
| BANKACTION | 2 | 0 | 0 |
| ACTIVITYSTATUS | 2 | 0 | 0 |
| ACTIVITYDISP | 2 | 0 | 0 |
| ACCOUNTSTATUS | 2 | 0 | 0 |

ACCOUNT STATUS

Owner: dbo
Target DB name: icns
Number of columns: 2
Number of Indexes: 0
Number of foreign keys: 0
Codes: 0
Type: 0

Extended Attributes:
OnFileGroup PRIMARY
Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|--------------------------------------|-------------|------------------------|-------------|
| STATUSID | tinyint | Not allowed | |
| DESCRIPTION | varchar(30) | Not allowed | |
| Foreign keys | | Child | Parent |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | | BANKACTIVITY.EPNFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_CHIPS | | BANKACTIVITY.CHIPSFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | | BANKACTIVITY.EPNFLAG | STATUSID |

| | | |
|--------------------------------------|------------------------|----------|
| FK_BANKACTIVITY-ACCOUNT STATUS_CHIPS | BANKACTIVITY.CHIPSFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | BANKACTIVITY.EPNFLAG | STATUSID |

Column details

1. STATUSID

Physical data type:

tinyint

Allow NULLs:

Not allowed

2. DESCRIPTION

Physical data type:

varchar(30)

Allow NULLs:

Not allowed

ACTIVITYDISP

Owner: dbo
Target DB name: icns
Number of columns: 2
Number of Indexes: 0
Number of foreign keys: 0
Codes: 0
Type: 0

Extended Attributes:
OnFileGroup PRIMARY
Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|--------------------------------------|-------------|------------------------|-------------|
| STATUSID | tinyint | Not allowed | |
| DESCRIPTION | varchar(30) | Not allowed | |
| Foreign keys | | Child | Parent |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | | BANKACTIVITY.EPNFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_CHIPS | | BANKACTIVITY.CHIPSFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | | BANKACTIVITY.EPNFLAG | STATUSID |

| | | |
|--------------------------------------|------------------------|----------|
| FK_BANKACTIVITY-ACCOUNT STATUS_CHIPS | BANKACTIVITY.CHIPSFLAG | STATUSID |
| FK_BANKACTIVITY-ACCOUNT STATUS_EPN | BANKACTIVITY.EPNFLAG | STATUSID |

Column details

1. STATUSID

Physical data type:

tinyint

Allow NULLs:

Not allowed

2. DESCRIPTION

Physical data type:

varchar(30)

Allow NULLs:

Not allowed

ACTIVITYSTATUS

Owner: dbo
Target DB name: icns
Number of columns: 2
Number of Indexes: 0
Number of foreign keys: 0
Codes: 0
Type: 0

Extended Attributes:
OnFileGroup PRIMARY
Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|---------------------------------|-------------|-----------------------------|-------------|
| STATUSID | tinyint | Not allowed | |
| DESCRIPTION | varchar(30) | Not allowed | |
| Foreign keys | | Child | Parent |
| FK_BANKACTIVITY_ACTIVITYSTATUS | | BANKACTIVITY.ACTIVITYSTATUS | STATUSID |
| FK_BANKACTIVITY-ACTIVITY STATUS | | UPOCACTIVITY.CHIPSFLAG | STATUSID |
| Column details | | | |

1. STATUSID

Physical data type:

tinyint

Allow NULLs:

Not allowed

2. DESCRIPTION

Physical data type:

varchar(30)

Allow NULLs:

Not allowed

BANKACTION

Owner: dbo
Target DB name: icns
Number of columns: 2,
Number of Indexes: 0
Number of foreign keys: 0
Codes: 0
Type: 0

Extended Attributes:
OnFileGroup PRIMARY
Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|---|-------------|----------------------|-------------|
| ACTIONID | tinyint | Not allowed | |
| DESCRIPTION | varchar(30) | Not allowed | |
| Foreign keys | | Child | Parent |
| FK_BANKACTIVITY_BANKACTIVITY_BANKACTION | | BANKACTIVITY.ACTIONS | ACTIONID |
| Column details | | | |

1. ACTIONID

| | |
|-----------------------|-------------|
| Physical data type: | tinyint |
| Allow NULLs: | Not allowed |
| <u>2. DESCRIPTION</u> | |
| Physical data type: | varchar(30) |
| Allow NULLs: | Not allowed |

BANKACTIVITY

Owner: dbo

Target DB name: icns

Number of columns: 41

Number of Indexes: 1

Number of foreign keys: 6

Codes: 3

Type: Table

User defined types:

| Name | Data Type |
|----------|------------------|
| BANKRTN | N-Signed Integer |
| CHIPSABA | N-Signed Integer |
| BANKRTN | N-Signed Integer |

Clustered PK

Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|---------------------|--------------|-------------|-------------|
| ACTIVITYID | Int Identity | Not allowed | |
| ACTIVITYDISP (FK) | tinyint | Not allowed | |
| APPLDATE | datetime | Allowed | |
| ACTIVITYSTATUS (FK) | tinyint | Not allowed | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------|-------------|-------------|-------------|
| CREATEDATE | datetime | Not allowed | |
| CREATEUSER | varchar(40) | Not allowed | |
| CREATESOURCE (FK) | tinyint | Not allowed | |
| APPRDATE | datetime | Allowed | |
| APPRUSER | varchar(40) | Allowed | |
| CANCELDATE | datetime | Allowed | |
| CANCELUSER | varchar(40) | Allowed | |
| ACTIONS (FK) | tinyint | Not allowed | |
| EFFDATE | datetime | Not allowed | |
| BANKRTN (II) | BANKRTN | Not allowed | |
| BANKNAME | varchar(96) | Not allowed | |
| EPNFLAG (FK) | tinyint | Not allowed | |
| CHIPSFALG (FK) | tinyint | Not allowed | |
| CHIPSABA | CHIPSABA | Allowed | |
| CHIPSLNKRTN | BANKRTN | Allowed | |
| CHIPSDDA | varchar(34) | Allowed | |

| Columns | Data Type | Allow NULLS | Value/Range |
|------------|-------------|-------------|-------------|
| CHIPSDEPTH | tinyint | Allowed | |
| BANKTYPE | varchar(3) | Not Allowed | |
| ADD1 | varchar(64) | Allowed | |
| ADD2 | varchar(64) | Allowed | |
| ADD3 | varchar(64) | Allowed | |
| ADD4 | varchar(50) | Allowed | |
| CITY | varchar(32) | Allowed | |
| STATE | varchar(32) | Allowed | |
| ZIP | varchar(11) | Allowed | |
| COUNTRY | char(3) | Not Allowed | |
| PHONE | Phone | Allowed | |
| FAX | Phone | Allowed | |
| EMAIL | email | Allowed | |
| BACKPHONE | Phone | Allowed | |
| BACKFAX | Phone | Allowed | |
| BACKEMAIL | email | Allowed | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|----------------|-------------------------|-------------|
| WPAYNA1 | varchar(35) | Allowed | |
| WPAYNA2 | varchar(35) | Allowed | |
| WPAYNA3 | varchar(35) | Allowed | |
| WPAYNA4 | varchar(35) | Allowed | |
| COMMENTS | varchar(80) | Allowed | |
| Indexes | Columns | Sort Order | |
| IX_BANKACTIVITY_BANKRTN (11) | BANKRTN | Ascending | |
| Foreign keys | Child | Parent | |
| FK_BANKACTIVITY_ACCOUNTSTATUS_EPN | EPNFLAG | ACCOUNTSTATUS.STATUSID | |
| FK_BANKACTIVITY_ACCOUNTSTATUS_CHIPS | CHIPSFLAG | ACCOUNTSTATUS.STATUSID | |
| FK_BANKACTIVITY_ACTIVITYDISP | ACTIVITYDISP | ACTIVITYDISP.DISPID | |
| FK_BANKACTIVITY_ACTIVITYSTATUS | ACTIVITYSTATUS | ACTIVITYSTATUS.STATUSID | |
| FK_BANKACTIVITY_BANKACTION | ACTIONS | BANKACTION.ACTIONID | |
| FK_BANKACTIVITY_SOURCE | CREATESOURCE | SOURCE.SOURCEID | |
| Column details | | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------|--------------|-------------|-------------|
| <u>1. ACTIVITYID</u> | | | |
| Physical data type: | int identity | | |
| Allow NULLs: | Not allowed | | |
| <u>2. ACTIVITYDISP (FK)</u> | | | |
| Physical data type: | tinyint | | |
| Allow NULLs: | Not allowed | | |
| <u>3. APPLDATE</u> | | | |
| Physical data type: | datetime | | |
| Allow NULLs: | Not allowed | | |
| <u>4. ACTIVITYSTATUS (FK)</u> | | | |
| Physical data type: | datetime | | |
| Allow NULLs: | Not allowed | | |
| <u>5. CREATEDATE</u> | | | |
| Physical data type: | datetime | | |
| Allow NULLs: | Not allowed | | |
| <u>6. CREATEUSER</u> | | | |
| Physical data type: | varchar(40) | | |
| Allow NULLs: | Not allowed | | |
| <u>7. CREATESOURCE (FK)</u> | | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|-------------------------|-------------|-------------|
| Physical data type: Allow NULLs: | tinyint Not allowed | | |
| <u>8. APPRDATE</u> | | | |
| Physical data type: Allow NULLs: | datetime Not allowed | | |
| <u>9. APPRUSER</u> | | | |
| Physical data type: Allow NULLs: | datetime Allowed | | |
| <u>10. CANCELDATE</u> | | | |
| Physical data type: Allow NULLs: | datetime Allowed | | |
| <u>11. CANCELUSER</u> | | | |
| Physical data type: Allow NULLs: | varchar(40) Allowed | | |
| <u>12. ACTIONS (FK)</u> | | | |
| Physical data type: Allow NULLs: | tinyint Not Allowed | | |
| <u>13. EFFDATE</u> | | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|---|--|-------------|-------------|
| Physical data type: Allow NULLs: Default value: | datetime Not allowed getdate() + 1 | | |
| <u>14. BANKRTN (I1)</u> | | | |
| Physical data type: Allow NULLs: | BANKRTN Not allowed | | |
| <u>15. BANKNAME</u> | | | |
| Physical data type: Allow NULLs: | varchar(96) Not allowed | | |
| <u>16. EPNFLAG (FK)</u> | | | |
| Physical data type: Allow NULLs: | tinyint Not allowed | | |
| <u>17. CHIPSFLAG (FK)</u> | | | |
| Physical data type: Allow NULLs: | tinyint Not allowed | | |
| <u>18. CHIPSABA</u> | | | |
| Physical data type: Allow NULLs: | CHIPSABA Allowed | | |
| <u>19. CHIPSINKRTN</u> | | | |
| | BANKRTN | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|----------------------------|-------------|-------------|
| Physical data type: Allow NULLs: | Allowed | | |
| <u>20. CHIPSDDA</u> | | | |
| Physical data type: Allow NULLs: | varchar(34) Allowed | | |
| <u>21. CHIPSDEPTH</u> | | | |
| Physical data type: Allow NULLs: | tinyint Allowed | | |
| <u>22. BANKTYPE</u> | | | |
| Physical data type: Allow NULLs: | varchar(3) Allowed | | |
| <u>23. ADD1</u> | | | |
| Physical data type: Allow NULLs: | varchar(64) Not Allowed | | |
| <u>24. ADD2</u> | | | |
| Physical data type: Allow NULLs: | varchar(64) Allowed | | |
| <u>25. ADD3</u> | | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|------------------------|-------------|-------------|
| Physical data type: Allow NULLs: | varchar(64) Allowed | | |
| <u>26. ADD4</u> | | | |
| Physical data type: Allow NULLs: | varchar(50) Allowed | | |
| <u>27. CITY</u> | | | |
| Physical data type: Allow NULLs: | varchar(32) Allowed | | |
| <u>28. STATE</u> | | | |
| Physical data type: Allow NULLs: | varchar(32) Allowed | | |
| <u>29. ZIP</u> | | | |
| Physical data type: Allow NULLs: | varchar(11) Allowed | | |
| <u>30. COUNTRY</u> | | | |
| Physical data type: Allow NULLs: | char(3) Not Allowed | | |
| <u>31. PHONE</u> | | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|---|------------------|-------------|-------------|
| Physical data type: Allow NULLs: <u>32. FAX</u> | Phone Allowed | | |
| Physical data type: Allow NULLs: <u>33. EMAIL</u> | Phone Allowed | | |
| Physical data type: Allow NULLs: <u>34. BACKPHONE</u> | email Allowed | | |
| Physical data type: Allow NULLs: <u>35. BACKFAX</u> | Phone Allowed | | |
| Physical data type: Allow NULLs: <u>36. BACKEMAIL</u> | Phone Allowed | | |
| Physical data type: Allow NULLs: <u>37. WPAYNA1</u> | email Allowed | | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|------------------------|-------------|-------------|
| Physical data type: Allow NULLs: | varchar(35) Allowed | | |
| <u>38. WPAYNA2</u> | | | |
| Physical data type: Allow NULLs: | varchar(35) Allowed | | |
| <u>39. WPAYNA3</u> | | | |
| Physical data type: Allow NULLs: | varchar(35) Allowed | | |
| <u>40. WPAYNA4</u> | | | |
| Physical data type: Allow NULLs: | varchar(35) Allowed | | |
| <u>41. COMMENTS</u> | | | |
| Physical data type: Allow NULLs: | varchar(80) Allowed | | |
| Code details | | | |

1. CK BANKACTIVITY ADD3

Type:

Check clause

Code body:

([ADD3] is null and [ADD2] is not null or [ADD3] is null and [ADD2] is null)

2. CK BANKACTIVITY ADD4

Type:

Check clause

Code body:

([ADD4] is null and [ADD3] is not null or [ADD4] is null and [ADD3] is null)

3. Checkactions

Type:

Trigger

Code body:

CREATE TRIGGER Checkactions ON dbo.BANKACTIVITY

FOR INSERT

AS

```
insert bankprofile
select bankrtn, bankname, createdate=getdate(),lastupdate=getdate(),epnflag,
chipsflag,
chipsaba, chipslnkrtn,chipsdda,chipsdepth,banktype,
add1,add2,add3,add4,city,state,zip,country,
phone,fax,email,backphone,backfax,backemail,
wpayna1,wpayna2,wpayna3,wpayna4 from inserted
where actions=1
```

Index details

IX BANKACTIVITY BANKRTN

| | |
|----------------------|---------------|
| Column(s): | BANKRTN (Asc) |
| Unique: | No |
| Extended attributes: | |
| OnFileGroup | F1 |
| CLUSTERED | No |
| IGNORE_DUP_KEY | No |
| FILLFACTOR | 0 |

PAD_INDEX No
DROP_EXISTING No
STATISTICS_NORECOMPUTE No

Foreign key details (child)

FK BANKACTIVITY ACCOUNTSTATUS EPN

Definition: Child Parent
 EPNFLAG ACCOUNTSTATUS.STATUSID

Relationship type:

Non-Identifying

Cardinality:

One-to-Zero-or-More

Allow NULLs:

Not allowed

Verb phrase:

is of

Ref. Integrity on update:

No action

Ref. Integrity on delete:

No action

FK BANKACTIVITY ACCOUNTSTATUS CHIPS

Definition: Child Parent
 CHIPSFLAG ACCOUNTSTATUS.STATUSID

| | |
|---------------------------|--|
| Relationship type: | Non-Identifying |
| Cardinality: | One-to-Zero-or-More |
| Allow NULLs: | Not allowed |
| Verb phrase: | hasFK_BANKACTIVITY_ACCOUNTSTATUS_CHIPS |
| Inverse phrase: | is of |
| Ref. Integrity on update: | No action |
| Ref. Integrity on delete: | No action |

FK BANKACTIVITY_ACTIVITYDISP

| | | |
|---------------------------|---------------------------------|---------------------|
| Definition: | Child | Parent |
| | ACTIVITYDISP | ACTIVITYDISP.DISPID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_BANKACTIVITY_ACTIVITYDISP | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

FK BANKACTIVITY ACTIVITYSTATUS CHIPS

| | | |
|---------------------------|---------------------------------|-------------------------------|
| Definition: | Child ACTIVITYDISP | Parent ACTIVITYDISP.DISPID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_BANKACTIVITY_ACTIVITYDISP | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

FK BANKACTIVITY ACTIVITYSTATUS

| | | |
|--------------------|-------------------------|---------------------------------|
| Definition: | Child ACTIVITYSTATUS | Parent ACTIVITYSTATUS.DISPID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |

| | | | |
|-----------------------------------|-----------------------------------|---------------------|--|
| Verb phrase: | hasFK_BANKACTIVITY_ACTIVITYSTATUS | | |
| Inverse phrase: | is of | | |
| Ref. Integrity on update: | No action | | |
| Ref. Integrity on delete: | No action | | |
| <u>FK BANKACTIVITY_BANKACTION</u> | | | |
| Definition: | Child | Parent | |
| Relationship type: | ACTIONS | BANKACTION.ACTIONID | |
| Cardinality: | Non-Identifying | | |
| Allow NULLs: | One-to-Zero-or-More | | |
| Verb phrase: | hasFK_BANKACTIVITY_BANKACTION | | |
| Inverse phrase: | is of | | |
| Ref. Integrity on update: | No action | | |
| Ref. Integrity on delete: | No action | | |
| <u>FK BANKACTIVITY_SOURCE</u> | | | |
| Definition: | Child | Parent | |

| | | |
|---------------------------|---------------------------|-----------------|
| | CREATESOURCE | SOURCE.SOURCEID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_BANKACTIVITY_SOURCE | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

FK BANKACTIVITY_SOURCE

| | | |
|---------------------------|---------------------------|-----------------|
| Definition: | Child | Parent |
| | CREATESOURCE | SOURCE.SOURCEID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_BANKACTIVITY_SOURCE | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |

No action

Ref. Integrity on delete:

BANKPROFILE

Owner: dbo
Target DB name: icns
Number of columns: 29
Number of Indexes: 0
Number of foreign keys: 3
Codes: 3
Type: Table
User defined types: Name

| | Data Type |
|----------|-----------------------|
| BANKRTN | |
| CHIPSABA | N-Signed Integer |
| BANKRTN | N-Signed Integer |
| Phone | N-Signed Integer |
| Phone | Integer |
| email | C-Variable Length(32) |
| Phone | C-Variable Length(32) |
| Phone | C-Variable Length(32) |

| | |
|-------|---------------------------|
| email | C-Variable Length(128) |
| | C-Variable Length(32) |
| | C-Variable Length(128) |
| | C-Variable Length(32) |
| | C-Variable Length(32) |
| | C-Variable Length(128) |

| | |
|----------------------|----|
| Extended Attributes: | |
| OnFileGroup | F1 |
| Clustered PK | No |

| Columns | Data Type | Allow NULLS | Value/Range |
|----------------|-------------|----------------|-------------|
| BANKRTN | BANKRTN | Not allowed | |
| BANKNAME | varchar(96) | Not allowed | |
| CREATEDATE | datetime | Not allowed | |
| LASTUPDT | datetime | Not allowed | |
| STATUS (FK) | tinyint | Not allowed | |
| EPNFLAG | tinyint | Not allowed | |
| CHIPSFLAG (FK) | tinyint | Not allowed | |
| CHIPSABA | chipsaba | Allowed | |
| CHIPSLINKRTN | bankrtn | Allowed | |
| CHIPSDDA | varchar(34) | Allowed | |
| CHIPSDEPTH | tinyint | Allowed | |
| ADD1 | varchar(64) | Not allowed | |
| ADD2 | varchar(64) | Allowed | |
| ADD3 | varchar(64) | Allowed | |
| ADD4 | varchar(64) | Allowed | |
| CITY | varchar(32) | Allowed | |

| | | | |
|------------------------------------|-------------|-------------|-------------------------|
| STATE | varchar(32) | Allowed | |
| ZIP | varchar(11) | Allowed | |
| COUNTRY | char(3) | Not allowed | |
| PHONE | Phone | Allowed | |
| FAX | Phone | Allowed | |
| EMAIL | email | Allowed | |
| BACKPHONE | Phone | Allowed | |
| BACKFAX | Phone | Allowed | |
| BACKEMAIL | email | Allowed | |
| WPAYNA1 | varchar(35) | Allowed | |
| WPAYNA2 | varchar(35) | Allowed | |
| WPAYNA3 | varchar(35) | Allowed | |
| WPAYNA4 | varchar(35) | Allowed | |
| Foreign keys | | Child | Parent |
| FK_BANKPROFILE_ACCOUNTSTATUS_EPM | | EPNFLAG | ACCOUNT.STATUSID |
| FK_BANKPROFILE_ACCOUNTSTATUS_CHIPS | | CHIPSFLAG | ACCOUNT.STATUSID |
| FK_BANKPROFILE_BANKSTATUS | | STATUS | BANKSTATUS.STATUS ID |

| FK_BANKROUTING_BANKPROFILE | BANROUT.LINKEDRTN | BANKRTN |
|---|--------------------------------------|---------|
| Column details | | |
| 1. <u>BANKRTN</u> Physical data type: Allow NULLs: | BANKRTN Not allowed | |
| 2. <u>BANKNAME</u> Physical data type: Allow NULLs | varchar(96) Not allowed | |
| 3. <u>CREATDATE</u> Physical data type: Allow NULLs Default value: | datetime Not allowed getdate() | |
| 4. <u>LASTUPDT</u> Physical data type: Allow NULLs Default value: | datetime Not allowed getdate() | |
| 5. <u>STATUS (FK)</u> Physical data type: Allow NULLs | tinyint Not allowed | |

| | |
|--|----------------------------|
| 6. <u>EPNFLAG (FK)</u> Physical data type: Allow NULLs | tinyint Not allowed |
| 7. <u>CHIPSFLAG (FK)</u> Physical data type: Allow NULLs | tinyint Not allowed |
| 8. <u>CHIPSABA</u> Physical data type: Allow NULLs | CHIPSABA Allowed |
| 9. <u>CHIPS LINKRTN</u> Physical data type: Allow NULLs | BANKRTN Allowed |
| 10. <u>CHIPSDDA</u> Physical data type: Allow NULLs | varchar(34) Allowed |
| 11. <u>CHIPSDEPTH</u> Physical data type: Allow NULLs | tinyint Allowed |
| 12. <u>ADD1</u> Physical data type: Allow NULLs | varchar(64) Not allowed |

13. ADD2

Physical data type:
Allow NULLs

varchar(64)
Allowed

14. ADD3

Physical data type:
Allow NULLs

varchar(64)
Allowed

15. ADD4

Physical data type:
Allow NULLs

varchar(64)
Allowed

16. CITY

Physical data type:
Allow NULLs

varchar(32)
Allowed

17. STATE

Physical data type:
Allow NULLs

varchar(32)
Allowed

18. ZIP

Physical data type:
Allow NULLs

varchar(11)
Allowed

19. COUNTRY

Physical data type:
Allow NULLs

char(3)
Not allowed

| | | |
|----------------------|------------------------------------|------------------------|
| <u>20. PHONE</u> | Physical data type: Allow NULLs | Phone Allowed |
| <u>21. FAX</u> | Physical data type: Allow NULLs | Phone Allowed |
| <u>22. EMAIL</u> | Physical data type: Allow NULLs | email Allowed |
| <u>23. BACKPHONE</u> | Physical data type: Allow NULLs | Phone Allowed |
| <u>24. BACKFAX</u> | Physical data type: Allow NULLs | Phone Allowed |
| <u>25. BACKEMAIL</u> | Physical data type: Allow NULLs | email Allowed |
| <u>26. WPAYNAI</u> | Physical data type: Allow NULLs | varchar(35) Allowed |

27. WPAYNA2

Physical data type:
Allow NULLs

varchar(35)
Allowed

28. WPAYNA3

Physical data type:
Allow NULLs

varchar(35)
Allowed

29. WPAYNA4

Physical data type:
Allow NULLs

varchar(35)
Allowed

Code details

1. CK BANK ADD3

Type:

Code body:

Check clause
([ADD3] is null and [ADD2] is not or [ADD3] is null
[ADD2] is null)

2. CK BANK ADD4

Type:

Code body:

Check clause
([ADD4] is null and [ADD3] is not null or [ADD4] is
null and [ADD3] is null)

3. NoModify Bank

Type: Trigger
Code body: CREATE TRIGGER NoModifyBank ON [dbo].
[BankProfile]

INSTEAD OF INSERT, UPDATE, DELETE

AS

Raiserror("Direct modification to this table is not
allowed", 16,1)

Foreign key details (child)

FK BANKPROFILE_ACCOUNTSTATUS_EPN

| | | |
|---------------------------|--------------------------------------|------------------------|
| Definition: | Child | Parent |
| | EPNFLAG | ACCOUNTSTATUS.STATUSID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | has FK_BANKPROFILE_ACCOUNTSTATUS_EPN | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

| <u>FK BANKPROFILE ACCOUNTSTATUS CHIPS</u> | | | |
|---|--|----------------------|--|
| Definition: | Child | Parent | |
| | CHIPSFLAG | ACCOUSTATUS.STATUSID | |
| Relationship type: | Non-Identifying | | |
| Cardinality: | One-to-Zero-or-More | | |
| Allow NULLs: | Not allowed | | |
| Verb phrase: | has FK_BANKPROFILE_ACCOUNTSTATUS_CHIPS | | |
| Inverse phrase: | is of | | |
| Ref. Integrity on update: | No action | | |
| Ref. Integrity on delete: | No action | | |

| <u>FK BANKPROFILE BANKSTATUS</u> | | | |
|----------------------------------|---------------------|---------------------|--|
| Definition: | Child | Parent | |
| | STATUS | BANKSTATUS.STATUSID | |
| Relationship type: | Non-Identifying | | |
| Cardinality: | One-to-Zero-or-More | | |
| Allow NULLs: | Not allowed | | |

| | |
|---------------------------|-------------------------------|
| Verb phrase: | has FK_BANKPROFILE_BANKSTATUS |
| Inverse phrase: | is of |
| Ref. Integrity on update; | No action |
| Ref. Integrity on delete: | No action |

BANKROUTING

| Owner: | dbo | | | | | | | | | |
|-------------------------|---|-------------|-------------|-------------|------------------|-------------|------------------|----------------|-------------|--|
| Target DB name: | icns | | | | | | | | | |
| Number of columns: | 2 | | | | | | | | | |
| Number of indexes: | 0 | | | | | | | | | |
| Number of foreign keys: | 1 | | | | | | | | | |
| Codes: | 0 | | | | | | | | | |
| Type: | Table | | | | | | | | | |
| User defined types: | <table><tr><th>Name</th><th>Data Type</th></tr><tr><td>BANKRTN</td><td>N-Signed Integer</td></tr><tr><td>BANKRTN</td><td>N-Signed Integer</td></tr></table> | Name | Data Type | BANKRTN | N-Signed Integer | BANKRTN | N-Signed Integer | | | |
| Name | Data Type | | | | | | | | | |
| BANKRTN | N-Signed Integer | | | | | | | | | |
| BANKRTN | N-Signed Integer | | | | | | | | | |
| Extended attributes: | | | | | | | | | | |
| OnFileGroup | PRIMARY | | | | | | | | | |
| Clustered PK | Yes | | | | | | | | | |
| Columns | <table><tr><th>Data Type</th><th>Allow NULLS</th><th>Value/Range</th></tr><tr><td>BANKRTN</td><td>Not allowed</td><td></td></tr><tr><td>LINKEDRTN (FK)</td><td>Not allowed</td><td></td></tr></table> | Data Type | Allow NULLS | Value/Range | BANKRTN | Not allowed | | LINKEDRTN (FK) | Not allowed | |
| Data Type | Allow NULLS | Value/Range | | | | | | | | |
| BANKRTN | Not allowed | | | | | | | | | |
| LINKEDRTN (FK) | Not allowed | | | | | | | | | |

| Foreign keys | | Child | Parent |
|-----------------------------------|---------------------|---------------------|---------------------|
| FK_BANKROUTING_BANKPROFILE | | LINKEDRTN | BANKPROFILE.BANKRTN |
| FK_PICACTIVITY_BANKROUTING | | PICACTIVITY.BANKRTN | BANKRTN |
| Column details | | | |
| 1. <u>BANKRTN</u> | | | |
| Physical data type: | | BANKRTN | |
| Allow NULLs: | | Not allowed | |
| 2. <u>LINKEDRTN (FK)</u> | | | |
| Physical data type: | | BANKRTN | |
| Allow NULLs: | | Not allowed | |
| Foreign key details (child) | | | |
| <u>FK BANKROUTING BANKPROFILE</u> | | | |
| Definition: | Child | Parent | |
| | LINKEDRTN | BANKPROFILE.BANKRTN | |
| Relationship type: | Non-Identifying | | |
| Cardinality: | One-to-Zero-or-More | | |

| | |
|---------------------------|-----------------------------------|
| Allow NULLs: | Not allowed |
| Verb phrase: | hasFK_BANKROUTING_BANKPROFIL E |
| Inverse phrase: | is of |
| Ref. Integrity on update: | No action |
| Ref. Integrity on delete: | No action |

BANKSTATUS

Owner: dbo
Target DB name: icns
Number of columns: 2
Number of indexes: 0
Number of foreign keys: 0
Codes: 0
Type: Table

Extended attributes:

OnFileGroup PRIMARY
Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|--------------|-------------|-------------|-------------|
| STATUS | tinyint | Not allowed | |
| DESCRIPTION | varchar(50) | Not allowed | |
| Foreign keys | | Child | Parent |

FK_BANKPROFILE_BANKSTATUS BANKPROFILE.STATUS STATUSID

| Column details | |
|---|----------------------------|
| 1. <u>STATUSID</u> Physical data type: Allow NULLs: | tinyint Not allowed |
| 2. <u>DESCRIPTION</u> Physical data type: Allow NULLs | varchar(50) Not allowed |

SOURCE

| | | | | |
|-------------------------|---------|-------------|---------------------------|-------------|
| Owner: | dbo | | | |
| Target DB name: | icns | | | |
| Number of columns: | 2 | | | |
| Number of Indexes: | 0 | | | |
| Number of foreign keys: | 0 | | | |
| Codes: | 0 | | | |
| Type: | Table | | | |
| Extended Attributes: | | | | |
| OnFileGroup | PRIMARY | | | |
| Clustered PK | Yes | | | |
| Columns | | Data Type | Allow NULLS | Value/Range |
| SOURCEID | | tinyint | Not allowed | |
| DESCRIPTION | | varchar(30) | Not allowed | |
| Foreign keys | | | Child | Parent |
| FK_BANKACTIVITY-SOURCE | | | BANKACTIVITY.CREATESOURCE | SOURCEID |
| FK_PICACTIVITY_SOURCE | | | PICACTIVITY.CREATESOURCE | SSOURCEID |
| Column details | | | | |

1. SOURCEID

Physical data type:
Allow NULLs:

tinyint
Not allowed

2. DESCRIPTION

Physical data type:
Allow NULLs:

varchar(30)
Not allowed

SYSTEMPARA

| | | | | |
|-------------------------|-----------|-------------|-------------|--|
| Owner: | dbo | | | |
| Target DB name: | icns | | | |
| Number of columns: | 2 | | | |
| Number of indexes: | 0 | | | |
| Number of foreign keys: | 0 | | | |
| Codes: | 0 | | | |
| Type: | Table | | | |
| Extended attributes: | | | | |
| OnFileGroup | PRIMARY | | | |
| Clustered PK | No | | | |
| Columns | Data type | Allow NULLs | Value/Range | |
| PICDIGITS | tinyint | Not allowed | | |
| PICSPEND | int | Not allowed | | |
| Column details | | | | |
| 1. <u>PICDIGITS</u> | | | | |

| | |
|-------------------------------------|------------------------|
| Physical data type: Allow NULLs: | tinyint Not allowed |
| <u>2. PICSPEND</u> | . |
| Physical data type: Allow NULLs: | int Not allowed |

PIC

| | | | | |
|-------------------------|---------|-----------------------|-------------|-------------|
| Owner: | dbo | | | |
| Target DB name: | icns | | | |
| Number of columns: | 31 | | | |
| Number of Indexes: | 1 | | | |
| Number of foreign keys: | 3 | | | |
| Codes: | 3 | | | |
| Type: | Table | | | |
| User defined types: | Name | Data Type | | |
| | PICTYPE | C-Variable Length(17) | | |
| | BANKRTN | N-Signed Integer | | |
| | PICTYPE | C-Variable Length(17) | | |
| | BANKRTN | N-Signed Integer | | |
| | Phone | C-Variable Length(32) | | |
| Extended attributes: | | | | |
| OnFileGroup | F1 | | | |
| Clustered PK | No | | | |
| Columns | | Data Type | Allow NULLS | Value/Range |

| | | |
|----------------|-------------|-------------|
| PICID | PICID | Not allowed |
| PICSTATUS (FK) | tinyint | Not allowed |
| EPNFLAG (FK) | tinyint | Not allowed |
| CHIPSFLAG (FK) | tinyint | Not allowed |
| BANKRTN (II) | BANKRTN | Not allowed |
| MASTERPIC | PICTYPE | Allowed |
| PICDDASIZE | tinyint | Not allowed |
| PICDDA | varchar(34) | Not allowed |
| CHIPSABA | smallint | Allowed |
| CHIPS LINKRTN | BANKRTN | Allowed |
| CORPNAME | varchar(96) | Not allowed |
| ADD1 | varchar(64) | Not allowed |
| ADD2 | varchar(64) | Allowed |
| ADD3 | varchar(64) | Allowed |
| ADD4 | varchar(64) | Allowed |
| CITY | varchar(32) | Allowed |
| STATE | varchar(32) | Allowed |
| ZIP | varchar(32) | Allowed |

| | | |
|---------------------|-------------|--------------|
| COUNTRY | char(3) | Not allowed |
| PHONE | Phone | Allowed |
| SICBCODE | smallint | Allowed |
| DUNSID | varchar(10) | Allowed |
| BICCODE | int | Allowed |
| BEICODE | smallint | Allowed |
| TAXID | int | Allowed |
| WPAYNA1 | varchar(35) | Allowed |
| WPAYNA2 | varchar(35) | Allowed |
| WPAYNA3 | varchar(35) | Allowed |
| WPAYNA4 | varchar(35) | Allowed |
| CREATDATE | datetime | Not allowed |
| LASTUPDT | datetime | Not allowed |
| Indexes | Columns | Sort order |
| IX_PIC_BANKRTN (11) | BANKRTN | Ascending |
| Foreign keys | | Child Parent |

| | | |
|-------------------------------|-----------|-----------------------|
| FK_BANKACCOUNT STATUS_EPN | EPNFLAG | PICACCOUNTSTATUSID |
| FK_PIC_PICACCOUNTSTATUS_CHIPS | CHIPSFLAG | PICACCOUNTSTATUSID |
| FK_PIC_PICSTATUS | PICSTATUS | PICSTATUS.PICSTATUSID |

Column details

1. PICID

Physical data type:
Allow NULLs:

PICTYPE
Not allowed

2. PICSTATUS (FK)

Physical data type:
Allow NULLs:

tinyint
Not allowed

3. EPNFLAG (FK)

Physical data type:
Allow NULLs:

tinyint
Not allowed

4. CHIPSFLAG (FK)

Physical data type:
Allow NULLs:

tinyint
Not allowed

5. BANKRTN (FK)

Physical data type:
Allow NULLs:

BANKRTN
Not allowed

| | |
|-------------------------|-------------|
| <u>6. MASTERRPIC</u> | |
| Physical data type: | PICTYPE |
| Allow NULLs: | Allowed |
| <u>7. PICDDASIZ</u> | |
| Physical data type: | tinyint |
| Allow NULLs: | Not allowed |
| <u>8. PICDDA</u> | |
| Physical data type: | varchar(34) |
| Allow NULLs: | Not allowed |
| <u>9. CHIPSABA</u> | |
| Physical data type: | smallint |
| Allow NULLs: | Allowed |
| Default value: | 0 |
| <u>10. CHIPSLINKRTN</u> | |
| Physical data type: | BANKRTN |
| Allow NULLs: | Allowed |
| <u>11. CORPNAME</u> | |
| Physical data type: | varchar(96) |
| Allow NULLs: | Not allowed |
| <u>12. ADD1</u> | |

| | |
|-------------------------------------|----------------------------|
| Physical data type: Allow NULLs: | varchar(64) Not allowed |
| <u>13. ADD2</u> | |
| Physical data type: Allow NULLs: | varchar(64) Not allowed |
| <u>14. ADD3</u> | |
| Physical data type: Allow NULLs: | varchar(64) Not allowed |
| <u>15. ADD4</u> | |
| Physical data type: Allow NULLs: | varchar(64) Allowed |
| <u>16. CITY</u> | |
| Physical data type: Allow NULLs: | varchar(32) Allowed |
| <u>17. STATE</u> | |
| Physical data type: Allow NULLs: | varchar(32) Not allowed |
| <u>18. ZIP</u> | |
| Physical data type: Allow NULLs: | varchar(32) Allowed |
| <u>19. COUNTY</u> | |

| | |
|-------------------------------------|------------------------|
| Physical data type: Allow NULLs: | char(3) Not allowed |
| 20. <u>PHONE</u> | |
| Physical data type: Allow NULLs: | Phone Allowed |
| 21. <u>SICBCODE</u> | |
| Physical data type: Allow NULLs: | Samalint Allowed |
| 22. <u>DUNSID</u> | |
| Physical data type: Allow NULLs: | varchar(10) Allowed |
| 23. <u>BICCODE</u> | |
| Physical data type: Allow NULLs: | int Allowed |
| 24. <u>BICCODE</u> | |
| Physical data type: Allow NULLs: | smallint Allowed |

25. TAXID

Physical data type:
Allow NULLs:

int
Allowed

26. WPAYNA1

Physical data type:
Allow NULLs:

varchar(35)
Allowed

27. WPAYNA2

Physical data type:
Allow NULLs:

varchar(35)
Allowed

28. WPAYNA3

Physical data type:
Allow NULLs:

varchar(35)
Allowed

29. WPAYNA4

Physical data type:
Allow NULLs:

varchar(35)
Allowed

30. CREATDATE

Physical data type:
Allow NULLs:
Default value:

datetime
Not allowed
getdate()

31. LASTUPDT

Physical data type:
Allow NULLs:
Default value:

datetime
Not allowed
getdate()

Code details

1. CK CORP ADD3

Type:
Code body:

Check clause
([ADD3] is null and [ADD2] is not null or [ADD3] is null and [ADD2] is null)

2. CK CORP ADD4

Type:
Code body:

Check clause
([ADD4] is null and [ADD3] is not null or [ADD4] is null and [ADD3] is null)

3. NoModifyPIC

Type:
Code body:

Trigger
CREATE TRIGGER NoModifyPIC ON [dbo].[PIC]

INSTEAD OF INSERT, UPDATE, DELETE

AS

Raiserror('Direct modification to this table is not allowed', 16,1)

Index details

| <u>IX PIC BANKRTN</u> | BANKRTN (Asc) |
|------------------------|---------------|
| Column(s): | |
| Unique: | No |
| Extended attributes: | |
| OnFileGroup | F2 |
| CLUSTERED | No |
| IGNORE_DUP_KEY | No |
| FILLFACTOR | 0 |
| PAD_INDEX | No |
| DROP_EXISTING | No |
| STATISTICS_NORECOMPUTE | No |

Foreign key details (child)

FK PIC PICACCOUNTSTATUS EPN

Foreign key details (child)

FK PIC PICACCOUNTSTATUS EPN

| | | |
|---------------------------|--------------------------------|---------------------------|
| Definition: | Child | Parent |
| | EPNFLAG | PICACCOUNTSTATUS.STATUSID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_PIC_PICACCOUNTSTATUS_EPN | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

FK PIC PICACCOUNTSTATUS CHIPS

| | | |
|--------------------|----------------------------------|---------------------------|
| Definition: | Child | Parent |
| | CHIPSFLAG | PICACCOUNTSTATUS.STATUSID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_PIC_PICACCOUNTSTATUS_CHIPS | |

| | | |
|---------------------------|---------------------|-----------------------|
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |
| <u>FK PIC PICSTATUS</u> | | |
| Definition: | Child | Parent |
| | PICSTATUS | PICSTATUS.PICSTATUSID |
| Relationship type: | Non-Identifying | |
| Cardinality: | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_PIC_PICSTATUS | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

PICACCOUNTSTATUS

Owner: dbo

Target DB name: icns

Number of columns: 2

Number of Indexes: 0

Number of foreign keys: 0

Codes: 0

Type: Table

Extended Attributes:

OnFileGroup PRIMARY

Clustered PK Yes

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------------------------|-------------|---------------------|-------------|
| STATUSID | tinyint | Not allowed | |
| DESCRIPTION | varchar(20) | Not allowed | |
| Foreign keys | | Child | Parent |
| FK_PIC_PICACCOUNTSTATUS_EPN | | PIC.EPNFLAG | STATUSID |
| FK_PICACCOUNTSTATUS_CHIPS | | PIC.CHIPSFLAG | STATUSID |
| FK_PICACTIVITY_PICACCOUNTSTATUS_EPN | | PICACTIVITY.EPNFLAG | STATUSID |

FK_PICACTIVITY_PICACCOUNTSTATUS_CHIPS PICACTIVITY.CHIPSFLAG STATUSID

Column details

1. STATUSID

Physical data type:
Allow NULLs:

tinyint
Not allowed

2. DESCRIPTION

Physical data type:
Allow NULLs:

varchar(20)
Not allowed

PICACTION

| | | | | | |
|--------------------------|---------|-------------|-----------------------|-------------|--|
| Owner: | dbo | | | | |
| Target DB name: | icns | | | | |
| Number of columns: | 2 | | | | |
| Number of Indexes: | 0 | | | | |
| Number of foreign keys: | 0 | | | | |
| Codes: | 0 | | | | |
| Type: | Table | | | | |
| Extended Attributes: | | | | | |
| OnFileGroup | PRIMARY | | | | |
| Clustered PK | Yes | | | | |
| Columns | | Data Type | Allow NULLS | Value/Range | |
| ACTIONID | | tinyint | Not allowed | | |
| DESCRIPTION | | varchar(20) | Not allowed | | |
| Foreign keys | | | Child | Parent | |
| FK_PICACTIVITY_PICACTION | | | PICACTIVITY.PICACTION | ACTIONID | |
| Column details | | | | | |

1. ACTIONID

Physical data type:
Allow NULLs:

tinyint
Not allowed

2. DESCRIPTION

Physical data type:
Allow NULLs:

varchar(20)
Not allowed

PICACTIVITY

| | | |
|-------------------------|----------|-----------------------|
| Owner: | dbo | |
| Target DB name: | icns | |
| Number of columns: | 41 | |
| Number of Indexes: | 2 | |
| Number of foreign keys: | 7 | |
| Codes: | 1 | |
| Type: | Table | |
| User defined types: | Name | Data Type |
| | PICTYPE | C-Variable Length(17) |
| | BANKRTN | N-Signed Integer |
| | CHIPSABA | N-Signed Integer |
| | BANKRTN | N-signed Integer |
| | Phone | C-Variable Length(32) |
| Extended attributes: | | |
| OnFileGroup | F2 | |
| Clustered PK | Yes | |

| Columns | Data Type | Allow NULLS | Value/Range |
|-------------------|--------------|-------------|-------------|
| ACTIVITYID | Int Identity | Not allowed | |
| ACTIVITYDISP (FK) | tinyint | Not allowed | |
| APPLDATE | datetime | Allowed | |
| STATUS (FK) | tinyint | Not allowed | |
| CREATEDATE | datetime | Not allowed | |
| CREATEUSER | varchar(40) | Not allowed | |
| CREATESOURCE (FK) | tinyint | Allowed | |
| APPRDATE | datetime | Allowed | |
| APPRUSER | varchar(40) | Allowed | |
| CANCELDATE | datetime | Allowed | |
| CANCELUSER | varchar(40) | Allowed | |
| PICACTION (FK) | tinyint | Not allowed | |
| EFFDATE | datetime | Allowed | |
| PICID (12) | PICTYPE | Allowed | |
| EFFDATE | tinyint | Not allowed | |
| PICID (12) | tinyint | Not allowed | |
| EPNFLAG (FK) | BANKRTN | Not allowed | |

| | | |
|------------------|-------------|-------------|
| CHIPSFLAG (FK) | tinyint | Not allowed |
| BANKRTN (FK, II) | BANKRTN | Not allowed |
| PICDDASIZE | tinyint | Not allowed |
| PICDDA | varchar(34) | Not allowed |
| CHIPSABA | CHIPSABA | Allowed |
| CHPSRTN | BANKRTN | Allowed |
| CORPNAME | varchar(96) | Not allowed |
| ADD1 | varchar(64) | Not allowed |
| ADD2 | varchar(64) | Allowed |
| ADD3 | varchar(64) | Allowed |
| ADD4 | varchar(64) | Allowed |
| CITY | varchar(32) | Allowed |
| STATE | varchar(32) | Allowed |
| ZIP | varchar(11) | Allowed |
| COUNTRY | char(3) | Not allowed |
| PHONE | phone | Allowed |
| SICBCODE | char(10) | Allowed |
| DUNSID | varchar(10) | Allowed |

| | | | |
|---------------------------------------|--------------|--------------|---------------------------|
| BICCODE | int | Allowed | |
| BEICODE | smallint | Allowed | |
| TAXID | int | Allowed | |
| WPAYNA1 | varchar(35) | Allowed | |
| WPAYNA2 | varchar(35) | Allowed | |
| WPAYNA3 | varchar(35) | Allowed | |
| WPAYNA4 | varchar(35) | Allowed | |
| COMMENTS | varchar(80) | Allowed | |
| Indexes | Columns | | Sort order |
| IX_PICACTIVITY_BANKRTN (11) | BANKRTN | | Ascending |
| IX_PICACTIVITY_PICID (12) | PICID | | Ascending |
| Foreign keys | | Child | Parent |
| FK_PICACTIVITY_ACTIVITYDISP | ACTIVITYDISP | ACTIVITYDISP | ACTIVITYDISP.DISPID |
| FK_PICACTIVITY_ACTIVITYSTATUS | STATUS | STATUS | ACTIVITYSTATUS.STATUSID |
| FK_PICACTIVITY_BANKROUTING | BANKRTN | BANKRTN | BANKROUTING.BANKRTN |
| FK_PICACTIVITY_SOURCE | CREATESOURCE | CREATESOURCE | SOURCE.SOURCEID |
| FK_PICACTIVITY_PICACCOUNTSTATUS_EPN | EPNFLAG | EPNFLAG | PICACCOUNTSTATUS.STATUSID |
| FK_PICACTIVITY_PICACCOUNTSTATUS_CHIPS | CHIPSFLAG | CHIPSFLAG | PICACCOUNTSTATUS.STATUSID |

| FK_PICACTIVITY_PICACTION | PICACTION | PICACTION.ACTIONID |
|------------------------------------|--------------|--------------------|
| Column details | | |
| 1. <u>ACTIVITYID</u> | | |
| Physical data type: | int identity | |
| Allow NULLs: | Not allowed | |
| 2. <u>ACTIVITYDISP (FK)</u> | | |
| Physical data type: | tinyint | |
| Allow NULLs: | Not allowed | |
| 3. <u>APPLDATE</u> | | |
| Physical data type: | datetime | |
| Allow NULLs: | Allowed | |
| 4. <u>STATUS (FK)</u> | | |
| Physical data type: | tinyint | |
| Allow NULLs: | Not allowed | |
| 5. <u>CREATEDATE</u> | | |
| Physical data type: | datetime | |
| Allow NULLs: | Not allowed | |

6. CREATE USER

Physical data type: varchar(40)
Allow NULLs: Not allowed

7. CREATESOURCE (FK)

Physical data type: tinyint
Allow NULLs: Allowed

8. APPRDATE

Physical data type: datetime
Allow NULLs: Allowed

9. APPRUSER

Physical data type: varchar(40)
Allow NULLs: Allowed

10. CANCELDATE

Physical data type: datetime
Allow NULLs: Allowed

11. CANCELUSER

Physical data type: varchar(40)
Allow NULLs: Allowed

12. PICACTION

Physical data type: tinyint
Allow NULLs: Not allowed

13. EFFDATE

Physical data type: datetime
Allow NULLs: Allowed
Default value: getdate() + 1

14. PICID (12)

Physical data type: PICTURE
Allow NULLs: Allowed

15. EPNFLAG (FK)

Physical data type: tinyint
Allow NULLs: Not allowed

16. CHIPSFLAG (FK)

Physical data type: tinyint
Allow NULLs: Not allowed

17. BANKRTN (FK, 11)

Physical data type: BANKRTN
Allow NULLs: Not allowed

18. PICDDASIZE

Physical data type: tinyint
Allow NULLs: Not allowed

19. PICDDA

| | |
|--------------------------------------|----------------------------|
| Physical data type: Allow NULLs: | varchar(34) Not allowed |
| <u>20. CHIPSABA</u> | |
| Physical data type: Allow NULLs:0 | CHIPSABA Allowed |
| <u>21. CHIPSRTN</u> | |
| Physical data type: Allow NULLs: | BANKRTN Allowed |
| <u>22. CORPNAME</u> | |
| Physical data type: Allow NULLs: | varchar(96) Not allowed |
| <u>23. ADD1</u> | |
| Physical data type: Allow NULLs: | varchar(64) Not allowed |
| <u>24. ADD2</u> | |
| Physical data type: Allow NULLs: | varchar(64) Allowed |
| <u>25. ADD3</u> | |
| Physical data type: Allow NULLs: | varchar(64) Allowed |
| <u>26. ADD4</u> | |

| | |
|-------------------------------------|------------------------|
| Physical data type: Allow NULLs: | varchar(64) Allowed |
| <u>27. CITY</u> | |
| Physical data type: Allow NULLs: | varchar(32) Allowed |
| <u>28. STATE</u> | |
| Physical data type: Allow NULLs: | varchar(32) Allowed |
| <u>29. ZIP</u> | |
| Physical data type: Allow NULLs: | varchar(11) Allowed |
| <u>30. COUNTRY</u> | |
| Physical data type: Allow NULLs: | char(3) Not allowed |
| <u>31. PHONE</u> | |
| Physical data type: Allow NULLs: | Phone Allowed |
| <u>32. SICBCODE</u> | |
| Physical data type: Allow NULLs: | char(10) Allowed |
| <u>33. DUNSID</u> | |

| | |
|-------------------------------------|------------------------|
| Physical data type: Allow NULLs: | varchar(10) Allowed |
| <u>34. BICCODE</u> | |
| Physical data type: Allow NULLs: | int Allowed |
| <u>35. BEICODE</u> | |
| Physical data type: Allow NULLs: | smallint Allowed |
| <u>36. TAXID</u> | |
| Physical data type: Allow NULLs: | int Allowed |
| <u>37. WPAYNA1</u> | |
| Physical data type: Allow NULLs: | varchar(35) Allowed |
| <u>38. WPAYNA2</u> | |
| Physical data type: Allow NULLs: | varchar(35) Allowed |

39. WPAYNA3

Physical data type: varchar(35)
Allow NULLs: Allowed

40. WPAYNA4

Physical data type: varchar(35)
Allow NULLs: Allowed

41. COMMENTS

Physical data type: varchar(80)
Allow NULLs: Allowed

Code details

1. AddPICActivity

Type:

Trigger

CREATE trigger dbo.AddPICActivity on dbo.PICACTIVITY

For INSERT

as

declare @action int

declare @PICID varchar (17)

Code body:

```
select @action=PICaction from inserted
if(@action=2)
if(notexists(select u.PICID from PIC u,insertd i where i.PICID=u.PICID))
Begin
raiserror ('PICID not exists, Transaction is rolled back',16,1)
ROLLBACK
End
if(NULL=(select PICID from inserted))
Begin
Raiserror('PICID cannot be NULL',16,1)
ROLLBACK
End
if(@action=1)
BEGIN
exec PickPIC @PICID OUT
Update PIC Activity
set PICID=@PICID
from PICTACTIVITY u, Inserted i
```

```
where u.activityID=i.activityID
insert PIC
select @PICID,2,epnflag,chipsflag,banktrn,null,PICDASIXE,PICDDA,
CHIPSABA,CHIPSRTN,Corpname,Add1,Add2,Add4,City,State,Zip,
Country,Phone,SICBCODE,DUNSID,BICCODE,BEICODE,TAXID,WPAYNA1,WPAY
NA2,WPAYNA3,WPAYNA4,Getdate(),Getdate() from Inserted
End
```

Index details

IX PICTACTIVITY BANKTRN

Columns(s)

Unique:

Extended attributes

OnFileGroup

CLUSTERED

IGNORE_DUP_KEY

FILLFACTOR

PAD_INDEX

DROP_EXISTING

BANKRPTN (Asc)

No

F1

No

No

0

No

No

STATISTICS_NORECOMPUTE

No

IX PICACTIVITY PICID

Column(s)

PICID (Asc)

Unique:

No

Extended attributes

OnFileGroup

F1

CLUSTERED

No

IGNORE_DUP_KEY

No

FILLFACTOR

0

PAD_INDEX

No

DROP_EXISTING

No

STATISTICS_NORECOMPUTE

No

Foreign key details (child)

FX PICACTIVITY_ACTIVITYDISP

Definition

Child

Parent

ACTIVITYDISP.DISPID

ACTIVITYDISP

Relationship type:

Cardinality

Allow NULLs:

Verb phrase:

Inverse phrase:

Ref. Integrity on update:

Ref. Integrity on delete:

Non-Identifying

One-to-Zero-or-More

Not allowed

hasFK_PICACTIVITY_ACTIVITYDISP

is of

No action

No action

FX PICTACTIVITY ACTIVITYSTATUS

Definition

Relationship type:

Cardinality

Allow NULLs:

Verb phrase:

Inverse phrase:

Ref. Integrity on update:

Child

STATUS

Non-Identifying

One-to-Zero-or-More

Not allowed

hasFK_PICACTIVITY_ACTIVITYSTATUS

is of

No action

Parent

ACTIVITYSTATUS-STATUSID

Ref. Integrity on delete:

No action

FK PICTACTIVITY BANKROUTING

Definition

Child

BANKRTN

Parent

BANKROUTING.BANKRTN

Relationship type:

Non-Identifying

Cardinality

One-to-Zero-or-More

Allow NULLs:

Not allowed

Verb phrase:

hasFK_PICTACTIVITY_BANKROUTING

Inverse phrase:

is of

Ref. Integrity on update:

No action

Ref. Integrity on delete:

No action

FK PICTACTIVITY SOURCE

Definition

Child

CREATESOURCE

Parent

SOURCE.SOURCEID

Relationship type:

Non-Identifying

Cardinality

Zero-or-One-to-Zero-or-More

| | | |
|--|--|----------------------------|
| Allow NULLs: | Allowed | |
| Verb phrase: | hasFK_PICACTIVITY_SOURCE | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |
| <u>FK PICTACTIVITY PICTACCOUNTSTATUS.EPN</u> | | |
| Definition | Child | Parent |
| | EPNFLAG | PICTACCOUNTSTATUS.STATUSID |
| Relationship type: | Non-Identifying | |
| Cardinality | One-to-Zero-or-More | |
| Allow NULLs: | Not allowed | |
| Verb phrase: | hasFK_PICTACTIVITY_PICTACCOUNTSTATUS-EPN | |
| Inverse phrase: | is of | |
| Ref. Integrity on update: | No action | |
| Ref. Integrity on delete: | No action | |

FK PICTACTIVITY PICTACCOUNTSTATUS CHIPS

| | | |
|---------------------------|---------------------------------------|---------------------------|
| Definition | Child | Parent |
| Relationship type: | CHIPSFLAG | PICACCOUNTSTATUS.STATUSID |
| Cardinality | Non-Identifying | |
| Allow NULLs: | One-to-Zero-or-More | |
| Verb phrase: | Not allowed | |
| Inverse phrase: | hasFK_PICACTIVITY_ACCOUNTSTATUS.CHIPS | |
| Ref. Integrity on update: | is of | |
| Ref. Integrity on delete: | No action | |

FK PICACTIVITY_PICACTION

| | | |
|--------------------|-----------------------------|--------------------|
| Definition | Child | Parent |
| Relationship type: | PICACTION | PICACTION.ACTIONID |
| Cardinality | Non-Identifying | |
| Allow NULLs: | One-to-Zero-or-More | |
| Verb phrase: | Not allowed | |
| Inverse phrase: | hasFK_PICACTIVITY_PICACTION | |
| | is of | |

Ref. Integrity on update: No action
Ref. Integrity on delete: No action

PICS

| | | | |
|-------------------------|-------------|-----------------------|-------------|
| Owner: | dbo | | |
| Target DB name: | ions | | |
| Number of columns: | 1 | | |
| Number of Indexes: | 0 | | |
| Number of foreign keys: | 0 | | |
| Codes: | 1 | | |
| Type: | Table | | |
| User defined types: | Name | Data Type | |
| | PICTYPE | C-Variable Length(17) | |
| Extended attributes: | | | |
| OnFileGroup | PRIMARY | | |
| Clustered PK | No | | |
| Columns | Data type | Allow NULLs | Value/Range |
| PICID | PICTYPE | Not allowed | |
| Column details | | | |
| 1. <u>PICID</u> | | | |
| Physical data type: | PICTYPE | | |
| Allow NULLs: | Not allowed | | |

Code details

1. UPDATESYPARA

Type:

Code body:

```
Trigger
CREATE TRIGGER DBO UPDATESYSPARA [dbo].[pics]
FOR INSERT
AS
UPDATE SYSTEMPARA
SET PICDIGITS=8,
PICSPEND=(SELECT
COUNT(*)FROM pics)
```

PICS

| | | | |
|-------------------------|---------------|-------------|-------------|
| Owner: | dbo | | |
| Target DB name: | icns | | |
| Number of columns: | 2 | | |
| Number of Indexes: | 0 | | |
| Number of foreign keys: | 0 | | |
| Codes: | 0 | | |
| Type: | Table | | |
| Extended attributes: | | | |
| OnFileGroup | PRIMARY | | |
| Clustered PK | Yes | | |
| Columns | Data type | Allow NULLs | Value/Range |
| PICSTATUSID | tinyint | Not allowed | |
| DESCRIPTION | varchar(30) | Not allowed | |
| Foreign keys | Child | | Parent |
| FK_PIC_PICSTATUS | PIC.PICSTATUS | | PICSTATUSID |

| Column details | |
|--|----------------------------|
| 1. <u>PICSTATUSD</u> Physical data type: Allow NULLs: | tinyint Not allowed |
| 2. <u>DESCRIPTION</u> Physical data type: Allow NULLs: | varchar(30) Not allowed |

[0089] The following is a list of application components associated with the PIC database. The list also shows functions associated with each component. This list is for illustrative purposes only.

- PIC
 - Add PIC
 - Input : PIC table fields, Initial Status of Activity record (approved or proposed)
 - Output: PIC number or error number, Activity ID number
 - Process: Ensure no PIC record and no PIC Activity record already exists for PIC. If no problems encountered, add an activity record to the PIC Activity table. If initial status is approved, assign PIC number and add record to PIC table as inactive. The PIC number will become active when the apply activity program is run for the PIC number's effective date.
 - Modify PIC
 - Input : PIC table fields, Initial Status of Activity record
 - Output: Error message if unsuccessful, Activity ID number
 - Process: If no problems exist, add activity record
 - Transfer PIC
 - Input : Required fields as indicated in business plan
 - Output: Error message if unsuccessful
 - Process: If no problems exist, add activity record.
 - Inquiry by DDA/RT (Demand Deposit Account/Routing Number)
 - Input : DDA number and RT
 - Output: A list of PIC records matching the input
 - Inquiry by PIC
 - Input : PIC number
 - Output: PIC record if found
 - Inquiry by Name

- Input : Name and/or address fields, routing number
- Output: List of possible matches
- Process: Only allow inquiry into PICs belonging to the requesting bank
- PIC Activity
 - Change Status of existing Activity
 - Input : Activity ID, New Status (approved, cancelled, applied etc.)
 - Output: Error message if unsuccessful
 - Inquire by Activity Request Number
 - Input : Activity ID
 - Output: Activity record
 - Inquire on all Activity in Proposed state for a bank
 - Input : Routing Number
 - Output: List of all proposed activity records
 - Inquire on all activity (closed and open) by a PIC number
 - Input : Routing Number, PIC number
 - Output: List of all activity records for PIC number
 - Process: Should not include activity records for that PIC number that were made by another bank in the case of a PIC that has gone through a transfer
- Bank Profile
 - Add Bank Profile Record
 - Input : All Bank Profile fields
 - Output: Error message if unsuccessful
 - Process: Set status to initialized so the first time a bank signs on to the web site it verifies all data, then set flag to active
 - Modify Bank Profile Record
 - Input : All Bank Profile fields

- Output: Error message if unsuccessful
- Process: If no problems encountered, add an activity record to the Bank Profile Activity table.
- Routing Number Table
 - Add routing number entry for a Master routing number
 - Input : Bank Routing Number, Master Bank Routing Number
 - Output: Error Message if unsuccessful
 - Delete routing number entry from a Master routing number
 - Input : Bank Routing Number, Master Bank Routing Number
 - Output: Error Message if unsuccessful
- Bank Profile Activity
 - Inquire by an RT number for all prior activity records
 - Input : Bank Routing Number, Master Bank Routing Number
 - Output: List of activity records
 - Users
 - Retrieve user info via active directory call
 - Add user via active directory call
 - Modify user attributes via active directory call
- Create program or use SQL (Structured Query Language) utility to apply and create a file of today's PIC and bank profile activity effective for the next business day and send this file to the CHIPS and EPN mainframes.
- Create program to process an incoming file from a bank and generate an outbound file. This will call the appropriate functions within the appropriate components above such as the Add PIC function. Each component needs to have functions for inquiring,

adding, modifying and deleting records. (No delete is required for PICs

[0090] For illustrative purposes, a number of PIC processes are depicted in tabular form in Tables 2 through 5. Each table provides the Name, Type, and Size of the Field, and depending on the type of process being performed whether the field is input/output of the process (X), an optional input/output (O), only displayed (D), or the result of the process (R).

[0091] Also, for illustrative purposes, the following is a list of potential business facade routines and business rules routines associated with the PIC database.

[0092] Business Facade Routines for PIC

[0093] GetPICbyPIC

[0094] Description: Get one the PICs with the specified PIC belonging to the User's RT. If a date is specified, show the historical representation for the requested date.

[0095] Parameters:

- PIC number
- Routing Transit Number
- Date (optional - only supply if looking for a specific date)

[0096] Output: PIC Dataset

[0097] GetPICbyDDART

[0098] Description: Get one or more PICs with the specified DDA/RT belonging to the User's RT. If a date is specified, show the historical representation for the requested date.

[0099] Parameters:

- DDA Number
- Routing Number
- User's Routing Number
- Date (optional - only supply if looking for a specific date) .

[0100] Output: Owner Flag: 1 if the PIC number belongs to user's bank PIC Data structure

| Table 2: PIC ACCOUNT PROCESS | | | | | | |
|------------------------------|----------------------|-------|------|-------------------|----------|--------------------------|
| | | ISSUE | PIC | MAINTAIN (MODIFY) | TRANSFER | INVESTIGATIONS (INQUIRY) |
| PIC ACCOUNT | PIC ACCOUNT | SZ | TYPE | | | |
| PIC IDENTIFIER | PIC IDENTIFIER | 17 | AN | R | D | D |
| PIC GLOBAL STATUS | PIC GLOBAL STATUS | 1 | N | X | X | D |
| EPN ACCOUNT STATUS | EPN ACCOUNT STATUS | 2 | N | X | X | D |
| CHIPS ACCOUNT STATUS | CHIPS ACCOUNT STATUS | 2 | N | X | X | D |
| BANK RT NUMBER | BANK RT NUMBER | 9 | N | X | X | D |
| PIC DDA SIZE | PIC DDA SIZE | 2 | N | X | X | D |
| PIC DDA | PIC DDA | 34 | AN | X | X | D |
| PIC CHIPS BANK ABA | PIC CHIPS BANK ABA | 4 | N | O | O | D |
| CHIPS LINK RT NUMBER | CHIPS LINK RT NUMBER | 9 | N | O | O | D |
| PIC BANK CATEGORY | PIC BANK CATEGORY | 2 | N | O | O | D |
| CORPORATE NAME | CORPORATE NAME | 96 | ANS | X | X | D |
| ADDRESS 1 | ADDRESS 1 | 64 | ANS | X | X | D |
| ADDRESS 2 | ADDRESS 2 | 64 | ANS | O | O | D |
| ADDRESS 3 | ADDRESS 3 | 64 | ANS | O | O | D |
| ADDRESS 4 | ADDRESS 4 | 64 | ANS | O | O | D |
| CITY | CITY | 32 | ANS | O | O | D |
| STATE/PROV | STATE/PROV | 32 | ANS | O | O | D |
| POSTAL CODE | POSTAL CODE | 11 | AN | O | O | D |
| COUNTRY | COUNTRY | 3 | ANS | X | X | D |
| PHONE NUMBER | PHONE NUMBER | 32 | AN | O | O | D |
| SIC BUSINESS CODE | SIC BUSINESS CODE | 4 | N | O | O | D |
| DUNS ID | DUNS ID | 10 | AN | O | O | D |
| BIC CODE | BIC CODE | 11 | N | O | O | D |
| BEI CODE | BEI CODE | 6? | N | O | O | D |
| TAX ID | TAX ID | 12 | N | O | O | D |
| WIRE PAY N/A 1 | WIRE PAY N/A 1 | 35 | ANS | O | O | D |
| WIRE PAY N/A 2 | WIRE PAY N/A 2 | 35 | ANS | O | O | D |
| WIRE PAY N/A 3 | WIRE PAY N/A 3 | 35 | ANS | O | O | D |

| | | | | | |
|----------------------|-----|----|---|---|---|
| WIRE PAY N/A 4 | ANS | 35 | O | O | D |
| CREATION DATETIME | N | 14 | R | R | D |
| LAST UPDATE DATETIME | N | 14 | R | R | D |

X - I/O of Process
O - Optimal I/O
D - Display Only
R- Result of Process

Table 3: PIC ACTIVITY PROCESS

| PIC ACTIVITY | TYPE | SZ | ISSUE PIC | MODIFY PIC | TRANSFER PIC | ACTIVITY INQUIRY | APPROVE ACTIVITY | REVISE ACTIVITY | CANCEL ACTIVITY |
|--------------------|------|----|--------------|---------------|-----------------|---------------------|---------------------|--------------------|--------------------|
| FIELD NAME | N | 12 | R | R | R | D | D | D | D |
| ACTIVITY ID | N | 1 | R | R | R | D | R | R | R |
| DISPOSITION | N | 14 | | | | D | | | |
| APPLIED DATETIME | N | 2 | R | R | R | D | R | R | R |
| ACTIVITY STATUS | N | 14 | R | R | R | D | D | D | D |
| CREATION DATETIME | AN | 40 | R | R | R | D | D | D | D |
| CREATED BY USER | N | 14 | | | | D | | | |
| APPROVAL | AN | 40 | | | | D | | | |
| DATETIME | N | 14 | | | | D | | | |
| APPROVED BY USER | AN | 40 | | | | D | R | R | R |
| CANCELLED | N | 14 | | | | D | | | |
| DATETIME | AN | 40 | | | | D | | | |
| CANCELLED BY USER | AN | 6 | R | R | R | D | D | X | D |
| PIC DETAIL ACTION | AN | 17 | R | R | R | D | D | X | D |
| PIC IDENTIFIER | N | 8 | R | R | R | D | D | O | D |
| UPDATE EFFECTIVE | N | 2 | R | R | R | D | D | X | D |
| DATE | N | 2 | R | R | R | D | D | X | D |
| EPN ACCOUNT | N | 2 | R | R | R | D | D | X | D |
| STATUS | N | 2 | R | R | R | D | D | X | D |
| CHIPS ACCOUNT | N | 9 | R | R | R | D | D | X | D |
| STATUS | N | 2 | R | R | R | D | D | X | D |
| BANK RT NUMBER | AN | 34 | R | R | R | D | D | X | D |
| PIC DDA SIZE | N | 4 | R | R | R | D | D | O | D |
| PIC DDA | N | 4 | R | R | R | D | D | O | D |
| PIC CHIPS BANK ABA | N | 4 | R | R | R | D | D | O | D |

| PIC ACTIVITY | TYPE | SZ | ISSUE PIC | MODIFY PIC | TRANSFER PIC | ACTIVITY INQUIRY | APPROVE ACTIVITY | REVISE ACTIVITY | CANCEL ACTIVITY |
|-------------------|------|----|--------------|---------------|-----------------|---------------------|---------------------|--------------------|--------------------|
| FIELD NAME | N | 9 | R | R | R | D | D | O | D |
| CHIPS LINK RT | | | | | | | | | |
| NUMBER | N | 2 | R | R | R | D | D | O | D |
| PIC BANK CATEGORY | ANS | 96 | R | R | R | D | D | X | D |
| CORPORATE NAME | ANS | 64 | R | R | R | D | D | X | D |
| ADDRESS 1 | ANS | 64 | R | R | R | D | D | O | D |
| ADDRESS 2 | ANS | 64 | R | R | R | D | D | O | D |
| ADDRESS 3 | ANS | 64 | R | R | R | D | D | O | D |
| ADDRESS 4 | ANS | 64 | R | R | R | D | D | O | D |
| CITY | ANS | 32 | R | R | R | D | D | O | D |
| STATE/PROV | ANS | 32 | R | R | R | D | D | O | D |
| POSTAL CODE | AN | 11 | R | R | R | D | D | O | D |
| COUNTRY | ANS | 3 | R | R | R | D | D | X | D |
| BIC CODE | N | 11 | R | R | R | D | D | O | D |
| BEI CODE | N | 6? | R | R | R | D | D | O | D |
| TAX ID | N | 12 | R | R | R | D | D | O | D |
| WIRE PAY N/A 1 | ANS | 35 | R | R | R | D | D | O | D |
| WIRE PAY N/A 2 | ANS | 35 | R | R | R | D | D | O | D |
| WIRE PAY N/A 3 | ANS | 35 | R | R | R | D | D | O | D |
| WIRE PAY N/A 4 | ANS | 35 | R | R | R | D | D | O | D |
| COMMENTS | | | R | R | R | D | D | | |

X - I/O of Process

O - Optimal I/O

D - Display Only

R- Result of Process

| Table 4: BANK PROFILE PROCESS | | | | | INVESTIGATIONS (INQUIRY) | |
|-------------------------------|----------|----------------|------|-----|--------------------------|---|
| BANK PROFILE | ADD BANK | MODIFY PROFILE | | | | |
| FIELD NAME | | | TYPE | SZ | | |
| BANK ROUTING NUMBER | X | X | N | 9 | | D |
| BANK STATUS | X | X | N | 1 | | D |
| BANK NAME | X | X | AN | 96 | | D |
| CREATION DATETIME | R | D | N | 14 | | D |
| LAST UPDATE DATETIME | R | R | N | 14 | | D |
| EPN FLAG | X | X | N | 1 | | D |
| CHIPS FLAG | X | X | N | 1 | | D |
| CHIPS BANK ABA | O | O | N | 4 | | D |
| CHIPS LINK RT NUMBER | O | O | N | 9 | | D |
| CHIPS DDA ACCT | O | O | AN | 34 | | D |
| CHIPS CHAIN DEPTH | O | O | N | 1 | | D |
| TYPE OF BANK | X | X | AN | 3 | | D |
| ADDRESS 1 | X | X | ANS | 64 | | D |
| ADDRESS 2 | O | O | ANS | 64 | | D |
| ADDRESS 3 | O | O | ANS | 64 | | D |
| ADDRESS 4 | O | O | ANS | 64 | | D |
| CITY | O | O | ANS | 32 | | D |
| STATE/PROV | O | O | ANS | 32 | | D |
| POSTAL CODE | O | O | AN | 11 | | D |
| COUNTRY | X | X | ANS | 3 | | D |
| PHONE NUMBER | O | O | AN | 32 | | D |
| FAX NUMBER | O | O | AN | 32 | | D |
| E-MAIL ADDRESS | O | O | ANS | 128 | | D |
| BACKUP PHONE NUMBER | O | O | AN | 32 | | D |
| BACKUP FAX NUMBER | O | O | AN | 32 | | D |
| BACKUP E-MAIL ADDRESS | O | O | ANS | 128 | | D |
| WIRE PAY N/A 1 | O | O | ANS | 35 | | D |
| WIRE PAY N/A 2 | O | O | ANS | 35 | | D |
| WIRE PAY N/A 3 | O | O | ANS | 35 | | D |
| WIRE PAY N/A 4 | O | O | ANS | 35 | | D |

X - I/O of Process
O - Optimal I/O
D - Display Only
R- Result of Process

Table 5: BANK PROFILE ACTIVITY PROCESS

| BANK PROFILE ACTIVITY | FIELD NAME | TYPE | SZ | ADD BANK PROFILE | MODIFY PROFILE | ACTIVIT Y | INQUIRY | APPROVE ACTIVIT Y | REVISE ACTIVITY | CANCEL ACTIVITY |
|-----------------------|-----------------------------|------|----|---------------------|-------------------|--------------|---------|-------------------------|--------------------|--------------------|
| | | | | | | | | | | |
| | ACTIVITY ID | N | 12 | R | R | D | D | D | D | D |
| | ACTIVITY DISPOSITION | N | 1 | R | R | D | D | D | R | R |
| | ACTIVITY APPLIED DATETIME | N | 14 | | | D | D | | | |
| | ACTIVITY STATUS | N | 2 | R | R | D | D | R | R | R |
| | ACTIVITY CREATION DATETIME | N | 14 | R | R | D | D | D | D | D |
| | CREATED BY USER | AN | 40 | R | R | D | D | D | D | D |
| | ACTIVITY APPROVAL DATETIME | N | 14 | | | D | D | R | | |
| | APPROVED BY USER | AN | 40 | | | D | D | R | | |
| | ACTIVITY CANCELLED DATETIME | N | 14 | | | D | D | R | | R |
| | CANCELLED BY USER | AN | 40 | | | D | D | R | R | R |
| | DETAIL ACTION | N | 1 | R | R | D | D | D | X | D |
| | UPDATE EFFECTIVE DATE | N | 8 | R | R | D | D | D | X | D |
| | BANK ROUTING NUMBER | N | 9 | R | R | D | D | D | D | D |
| | BANK STATUS | N | 1 | R | R | D | D | D | X | D |
| | BANK NAME | AN | 96 | R | R | D | D | D | X | D |
| | CREATION DATETIME | N | 14 | R | R | D | D | D | D | D |
| | LAST UPDATE DATETIME | N | 14 | R | R | D | D | D | R | D |
| | EPN FLAG | N | 1 | R | R | D | D | D | X | D |
| | CHIPS FLAG | N | 1 | R | R | D | D | D | X | D |
| | CHIPS BANK ABA | N | 4 | R | R | D | D | D | O | D |
| | CHIPS LINK RT NUMBER | N | 9 | R | R | D | D | D | O | D |
| | CHIPS DDA ACCT | AN | 34 | R | R | D | D | D | O | D |
| | CHIPS CHAIN DEPTH | N | 1 | R | R | D | D | D | O | D |
| | TYPE OF BANK | AN | 3 | R | R | D | D | D | X | D |
| | ADDRESS 1 | ANS | 64 | R | R | D | D | D | X | D |
| | ADDRESS 2 | ANS | 64 | R | R | D | D | D | O | D |
| | ADDRESS 3 | ANS | 64 | R | R | D | D | D | O | D |

| BANK PROFILE ACTIVITY | FIELD NAME | TYPE | SZ | ADD BANK PROFILE | MODIFY PROFILE | ACTIVIT | | APPROVE ACTIVIT | REVISE ACTIVITY | CANCEL ACTIVITY |
|-----------------------|-----------------------|------|-----|---------------------|-------------------|---------|---------|--------------------|--------------------|--------------------|
| | | | | | | Y | INQUIRY | Y | | |
| | ACTIVITY ID | N | 12 | R | R | D | D | D | D | D |
| | ADDRESS 4 | ANS | 64 | R | R | D | D | D | O | D |
| | CITY | ANS | 32 | R | R | D | D | D | O | D |
| | STATE/PROV | ANS | 32 | R | R | D | D | D | O | D |
| | POSTAL CODE | AN | 11 | R | R | D | D | D | O | D |
| | COUNTRY | ANS | 3 | R | R | D | D | D | X | D |
| | PHONE NUMBER | AN | 32 | R | R | D | D | D | O | D |
| | FAX NUMBER | AN | 32 | R | R | D | D | D | O | D |
| | E-MAIL ADDRESS | ANS | 128 | R | R | D | D | D | O | D |
| | BACKUP PHONE NUMBER | AN | 32 | R | R | D | D | D | O | D |
| | BACKUP FAX NUMBER | AN | 32 | R | R | D | D | D | O | D |
| | BACKUP E-MAIL ADDRESS | ANS | 128 | R | R | D | D | D | O | D |
| | WIRE PAY N/A 1 | ANS | 35 | R | R | D | D | D | O | D |
| | WIRE PAY N/A 2 | ANS | 35 | R | R | D | D | D | O | D |
| | WIRE PAY N/A 3 | ANS | 35 | R | R | D | D | D | O | D |
| | WIRE PAY N/A 4 | ANS | 35 | R | R | D | D | D | O | D |

X - I/O of Process

O - Optimal I/O

D - Display Only

R - Result of Process

[0101] GetPICbyCorpNameCity

[0102] Description: Get all PICs that begin with the Corporation Name supplied and restrict by the city name if supplied. No date is allowed for this search.

[0103] Parameters:

- Corp Name
- City Name

[0104] Output: PIC dataset of all PICs meeting the search

[0105] CreatePIC

[0106] Description: Create a PIC record. Sets up a PIC Dataset and calls the business rules' insert PIC procedure which validates all fields passed. This routine is passed a flag saying whether this is just a validate call or a validate and update call. Boolean procedure returning true if successful.

[0107] Parameters:

- All PIC fields
- Routing Number of User requesting the create
- Update or Validate only

[0108] GetPICtoModify

[0109] Description: Gets a PIC record to modify. If PIC is valid, then it returns the PIC data record from the PIC table. If open activity exists, the activity id is set to the activity ID of the pending activity. Boolean procedure returning true if successful.

[0110] Parameters:

- PIC
- Routing Number of User requesting the create
- PICdata

[0111] UpdatePIC

[0112] Description: Modify a PIC record. Passes a PIC dataset to the business rules' update PIC procedure which validates all fields passed. This routine is

passed a flag saying whether this is just a validate call or a validate and update call.
Boolean procedure returning true if successful.

[0113] Parameters:

- All PIC fields in PIC dataset form
- Update or Validate only

[0114] ClosePIC

[0115] Description: Changes the status of a PIC record to closed. Passes the PIC Dataset to the business rules' closethePIC procedure which validates all fields passed. Boolean procedure returning true if successful.

[0116] Parameters:

- PIC data

[0117] ReactivatePIC

[0118] Description: Changes the status of a closed PIC record to open. Passes the PIC Dataset to the business rules' close PIC procedure which validates all fields passed. Boolean procedure returning true if successful.

[0119] Parameters:

- PIC data

[0120] ApproveActivity

[0121] Description: Changes the status of a proposed PIC activity record to approved. Calls the business rules' approve PIC Activity procedure. Boolean procedure returning true if successful.

[0122] Parameters:

- PIC data set including the activity ID number and the user name approving the record

[0123] CancelActivity

[0124] Description: Changes the status of an open PIC activity record to canceled. Calls the business rules' cancel PIC Activity procedure. Boolean procedure returning true if successful.

[0125] Parameters:

- PIC data set including the activity ID number and the user name approving the record.

[0126] TransferPIC

[0127] Description: Transfers a PIC from one bank to another. Calls the business rules' transfer PIC procedure. Boolean procedure returning true if successful.

[0128] Parameters:

- PIC dataset including the PIC, Requesting User's RT, the new RT, new DDA numjber and new TaxID

[0129] ContestTransfer

[0130] Description: Contests a PIC Transfer. Calls the business rules' contest transfer procedure. Boolean procedure returning true if successful.

[0131] Parameters:

- PIC data set including the activity ID number and user contesting the transfer

[0132] Release Transfer

[0133] Description: Releases a PIC Transfer. Calls the business rules' release transfer procedure. Boolean procedure returning true if successful.

[0134] Parameters:

- PIC dataset including the activity ID and the user releasing the transfer

[0135] Business Facade Routines for URIC Activity

[0136] SearchActivitybyPIC

[0137] Description: Get all PIC activity for a given PIC. Routing number is supplied to ensure the PIC belongs to the requestor. If a date is supplied restrict the selection to be between the dates given.

[0138] Parameters:

- PIC
- Routing Transit Number
- Optional Start Date

- Optional End Date

[0139] Output: PIC dataset of all PIC activity records of activity records

[0140] SearchActivityByType

[0141] Description: Get all PIC activity for a bank based on the type of activity.

Routing number is supplied to ensure the PIC belongs to the requestor. If a date is supplied restrict the selection to be between the dates given.

[0142] Parameters:

- Activity Type
- Routing Transit Number
- Optional Start Date

[0143] Optional End Date Output: PIC dataset of all URIC activity records of activity records

[0144] SearchActivityByStatus

[0145] Description: Get all PIC activity for a bank based on a given activity status.

Routing number is supplied to ensure the PIC belongs to the requestor. If a date is supplied restrict the selection to be between the dates given.

[0146] Parameters:

- Activity Status
- Routing Transit Number
- Optional Start Date
- Optional End Date

[0147] Output: URIC dataset of all PIC activity records of activity records

[0148] SearchActivitybyCorpName

[0149] Description: Get all PIC activity for a bank based on a company name.

Routing number is supplied to ensure the PIC belongs to the requestor. If a date is supplied restrict the selection to be between the dates given.

[0150] Parameters:

- Company Name
- Routing Transit Number

- Optional Start Date

[0151] Optional End Date Output: PIC dataset of all PIC activity records of activity records

[0152] SearchActivitybyUserName

[0153] Description: Get all PIC activity for a bank created by a specific user.

Routing number is supplied to ensure the PIC belongs to the requestor. If a date is supplied restrict the selection to be between the dates given.

[0154] Parameters:

- User Name
- Routing Transit Number
- Optional Start Date
- Optional End Date

[0155] Output: PIC dataset of all PIC activity records of activity records

[0156] GetPICActivitybyID

[0157] Description: Get a specific PIC activity record

[0158] Parameters:

- Activity ID Number

[0159] Output: PIC dataset of all PIC activity records of activity records

[0160] Business Rules Routines for PIC

[0161] Insert

[0162] Description: Validates all fields passed . If update is requested and all edits pass, the new PIC is created Boolean procedure returning true if successful.

[0163] Validates the following

- All fields for proper size and character type
- DDA must be 17 characters or less if EPN enabled. If only CHIPS enabled it can be up to 35
- Must be either CHIPS enabled or EPN enabled
- If bank is not CHIPS enabled, CHIPS enabled flag cannot be on
- If bank is not EPN enabled, EPN enabled flag cannot be on

- Cannot skip an address line
- If CHIPS enabled, Wire pay name and address is required
- Cannot skip a wire pay name and address line
- Must be a valid business date. Cannot be less than the next valid effective date (either tomorrow or the next day if the update has already been run)
- RTN must belong to the bank requesting
- DDA/RT combination cannot already exist

[0164] Parameters:

- All PIC fields via the PIC dataset
- Routing Number of User requesting the create
- Update or Validate only

[0165] Update

[0166] Description: Validates all fields passed. If validate only is requested, only validation is done. If all fields pass all edits and update is requested, the PIC record is updated by creating an activity record. Boolean procedure returning true if successful.

[0167] Validates all the same as Insert with the following exceptions:

- DA/RT can already exist
- Must notify if open activity already exists for the PIC

[0168] Parameters:

- All PIC fields via the PIC dataset
- Update or Validate only

[0169] ClosethePIC

[0170] Description: Changes the status of a PIC record to closed. Validates the close request. If passes the edits, calls the update data access routing to create an activity record with an action type of close. Boolean procedure returning true if successful.

[0171] Parameters:

- PIC data set

[0172] Validates: effective date for a valid business date

[0173] Reactivate the PIC

[0174] Description: Changes the status of a PIC record to open. Validates the reactivate request. If passes the edits, calls the update data access routing to create an activity record with an action type of reactivate. Boolean procedure returning true if successful.

[0175] Parameters:

- PIC data set

[0176] Validates: effective date for a valid business date

[0177] Approve Activity

[0178] Description: Changes the status of a proposed PIC activity record to approved if validation is successful. Calls the data access Update Status routine. Boolean procedure returning true if successful.

[0179] Parameters:

- Activity ID Number

[0180] Validates: The user approving is not the user that entered the activity

[0181] Cancel Activity

[0182] Description: Changes the status of an open PIC activity record to canceled if the validation is successful. Calls the data access Update Status routine. Boolean procedure returning true if successful.

[0183] Parameters:

- Activity ID Number

[0184] Validates: The user canceling the activity is either the creator or an administrator

[0185] Transfer PIC

[0186] Description: Transfers a PIC from one bank to another if all validations are successful. Boolean procedure returning true if successful.

[0187] Parameters:

- PIC data as dataset

[0188] Validates:

- Effective date is valid business date
- Must be either EPN capable or CHIPS capable
- Cannot be EPN capable if bank is not EPN capable
- Cannot be CHIPS capable if bank is not CHIPS capable
- DDA number must be alphanumeric
- DDA number cannot exceed 17 characters if EPN capable

[0189] ContestTransfer

[0190] Description: Updates the PIC activity record with the status of contested. Calls the data access Update Status routine. Boolean procedure returning true if successful.

[0191] Parameters:

- Activity ID Number
- User contesting the transfer

[0192] ReleaseTransfer

[0193] Description: Updates the PIC activity record with the status of released. Calls the data access Update Status routine. Boolean procedure returning true if successful.

[0194] Parameters:

- Activity ID Number
- User releasing the transfer

[0195] Data Access Routines for PIC

[0196] FindPICbyPIC

[0197] Description: Gets a PIC record from PIC Activity based on a date passed. The routine passes along a flag which indicates whether the requesting bank has the right to see the entire PIC information or just a limited view. Boolean procedure returning true if successful.

[0198] Parameters:

- Routing Number of User requesting the find
- PIC
- Date

[0199] Stored Procedure Called: pfindpic_picid

[0200] FindPICbyAction

[0201] Description: Get a PIC activity record by the activity ID number

[0202] Parameters:

- Activity ID Number

[0203] Stored Procedure Called: pviewwpicactivity

[0204] FindPICbyComCity

[0205] Description: Get a list of PICs by querying on Company name and city name

[0206] Parameters:

- Routing Number of User requesting the find
- Corporate Name
- City Name

[0207] Stored Procedure Called: pfindpic_name

[0208] FindPICbyRTNDDA

[0209] Description: Retrieve a PIC by Routing Number/DDA number for a specific date. The routing passes along a flag which indicates whether the requestor has the right to see the entire PIC information or just a limited view.

[0210] Parameters:

- Routing Number of User requesting the find
- Routing Number of the PIC
- DDA Number of the PIC
- Date

[0211] ValidateEffectiveDate

[0212] Description: Validate if a specific date is a business date. Also must be within 90 days.

[0213] Parameters:

- Effective Date

[0214] Stored Procedure Called: pvalidateRTN

[0215] ValidateRTN

[0216] Description: Validates whether a routing number belongs to the user's bank

[0217] Parameters:

- Routing Number of requesting User
- Routing Number to verify

Stored Procedure Called: pvalidateRTN

[0218] ValidateRTDDAUnique

[0219] Description: Verifies the Routing Number/DDA Number combination does not already exist for any PIC,

[0220] Parameters:

- DDA Number
- Routing Number

[0221] Stored Procedure Called: pvalidatertn_DDA

[0222] GetActivitybyCorpName

[0223] Description: Gets all PIC activity records for a given corporate name. If dates are supplied, the records are restricted between the dates. Records are also restricted based on the user RTN.

[0224] Parameters:

- RTN of the user requesting the search
- Corporate Name
- Optional Start date and End date

[0225] Stored Procedure Called: pfindUA_Name

[0226] GetActivitybyStatus

[0227] Description: Gets all PIC activity records with a given activity status. If dates are supplied, the records are restricted between the dates. Records are also restricted based on the user RTN.

[0228] Parameters:

- RTN of the user requesting the search
- Activity Status
- Optional Start date and End date

[0229] Stored Procedure Called: pfindUA_Status

[0230] GetActivitybyType

[0231] Description: Gets all PIC activity records with a given activity type. If dates are supplied, the records are restricted between the dates. Records are also restricted based on the user RTN.

[0232] Parameters:

- RTN of the user requesting the search
- Activity Type
- Optional Start date and End date

[0233] Stored Procedure Called: pfindUA_Type

[0234] GetActivitybyPIC

[0235] Description: Gets all PIC activity records for a given PIC., If dates are supplied, the records are restricted between the dates. Records are also restricted based on the user RTN.

[0236] Parameters

- RTN of the user requesting the search
- PIC ID Number
- Optional Start date and End date

[0237] Stored Procedure Called: pfindUA_PICID

[0238] GetActivitybyUserName

[0239] Description: Gets all PIC activity records created by a specific user. If dates are supplied, the records are restricted between the dates. Records are also restricted based on the user RTN.

[0240] Parameters:

- RTN of the user requesting the search
- User Name
- Optional Start date and End date

[0241] Stored Procedure Called: pfindUA_User

[0242] A functional process model illustrating an approach to gather business requirements is depicted in Figure 7. In developing the process model, potential PIC functionality categories from registration to reporting are defined.

[0243] The sections that follow outline the detailed functional requirements related to the PIC to the framework outlined in Figure 7.

Register Bank Participants

[0244] Business customers who use the PIC functionality must have DDAs at a bank registered with the trusted third party to distribute PICs. Banks participating in the system are the primary channels for reaching business customers for all PIC activities, from marketing to maintenance. Preferably, no direct interaction between business customers and the system support team is expected. One of the advantages of the implementation of the PIC is to provide services to banks that transparently enhance bank relationships with their customers.

Create Bank Profile

[0245] To establish a PIC system relationship with a trusted third party, individual banks preferably complete a formal registration process. Formal registration requires each bank to provide entity-type information about itself. The trusted third party uses this information to create a bank profile, which is stored as part of the system database. Bank profiles serve as the foundation for providing customer

service for the complement of system enhancements. The information stored in the bank profile includes: bank name, primary address, primary contact, EPN registered, CHIPS registered, etc.

[0246] In addition, the profile includes information related to individual bank preferences. Examples of preference information include requirements related to maintenance approval and preferred data format(s). Furthermore, if a bank uses service providers to maintain individual DDAs, then a list of approved service providers authorized to access and maintain PIC data on a member bank's behalf is stored in the bank profile. The bank profile is flexible and capable of accommodating additional data elements.

Validate Bank Profile

[0247] Individual bank profile information is collected and validated prior to issuing PIC numbers to customer DDAs. After profile information is received, the system performs a validation process, checking all elements of the bank profile for accuracy.

[0248] Initially, all banks registering for the system are to be current participants of CHIPS or EPN. After initial deployment, banks are able to register for services without being CHIPS or EPN participants. However, such banks, who would typically be correspondents of CHIPS or EPN participants, must complete a bank profile with the system to gain access to PIC administrative functions. Also, non-participant banks must specify how transactions are to be executed, either through correspondent relationships or the Federal Reserve. Existing CHIPS or EPN system participants are not required to execute additional agreements for PIC. For current participants, the trusted third party expedites the registration process by leveraging information from the EPN and/or CHIPS platforms to automatically complete individual bank profiles. In this situation, the registering bank is required to supply any missing profile information and confirm the auto-populated profile data elements.

Build Initial PIC User Base

[0249] Once bank registration is complete, a bank is eligible to begin the process of requesting PICs for existing DDAs. To promote extensive participation and to accelerate PIC adoption, the system would preferably mass enroll all appropriate participating bank business customer DDAs in the PIC program. Mass enrollment, or mass PIC issuance, preferably takes place via bulk file transfer. Bulk file transfer is described later.

Perform Mass Enrollment

[0250] Participating bank business customers are automatically assigned a PIC for each of their DDAs via a mass enrollment process. The mass enrollment process relies on customer DDA data supplied by participating banks to populate the PIC database. Data is supplied by participating banks via a database extract file. The specifications for the database extract file are standard and defined by the system. Data fields required to complete DDA mass enrollment are identical to data fields required to complete a single DDA PIC enrollment. The database extract file elements preferably include: DDA name, address, account type, routing number, account number, etc.

[0251] Only information required for PIC creation is included in the mass enrollment requirements. Sensitive information (e.g., credit scoring) is not a PIC requirement.

[0252] Participant banks would preferably request PIC numbers only for DDAs that receive remittance payments. As a result, banks should identify and exclude non-remittance accounts from the mass enrollment file. The system preferably accepts files via ConnectDirect (only for banks who currently have software installed and frame-relay links to the trusted third party), in cases where banks do not have connectivity via SWIFTNet. Files may also be accepted via the Internet. In order to make the mass enrollment process simple, the system does not require

changes to current bank DDA systems. Upon completion of the mass PIC enrollment process, the system returns a file containing all PIC numbers and their respective DDA information to participating banks. The PIC file is returned to banks in the same data format in which it was received. When format or data omission errors occur in the mass enrollment file, the individual items in error are returned to the participating bank with a reject code. The entire enrollment file is not rejected unless significant format problems exist.

Issue PICs

Request PIC

[0253] PIC numbers are issued for business customer accounts by the system upon request from a registered bank or an approved service provider. Banks who are not system registered cannot request PICs. Also, a bank (or approved service provider) can request PICs for their business customer accounts only. While business customers actually own PICs, the requesting bank (or approved service provider) is responsible for the PIC until the PIC is closed or transferred to another bank. PIC requesting banks are required to provide the information necessary for PIC creation. PIC responsible banks also coordinate all PIC maintenance for related customer and correspondent bank accounts. Participating banks can request PIC numbers through multiple channels. These channels include: Batch file request; real-time request via web-site; and real-time request via messaging.

[0254] Individual PIC creation requests are initiated via a system website or messaging over SWIFTNet. Such transactions may also be executed via the Internet.

Assign PIC

[0255] Upon receipt of a PIC assignment request, the system checks the requesting DDA number to determine if it has been previously issued a PIC. If a PIC already exists for a given DDA number, the assignment request is rejected. Otherwise, the system creates a new record and assigns a PIC. The system issues a single PIC for each DDA number. A PIC is the same for a DDA whether transactions occur through EPN or CHIPS. Unlike PIC numbers, system routing transit numbers will differ for the CHIPS and EPN platforms, or any other platforms that may be supported for PICs. However, the system allows multiple PIC numbers to point to a single DDA to accommodate cases where companies merge or bank architecture changes.

[0256] PIC numbers are assigned randomly from a given block of numbers determined by system administrators. PIC numbers range from a minimum of eight to a maximum of 17 numeric characters. Initially, eight digit numbers are assigned -- six digits plus two check digits. The two right-most digits comprise the two check digits. The six left-most digits in an eight-digit PIC are used to compute the check digits.

[0257] The two check digits are employed to mitigate transposition errors and ensure that each PIC numbers are unique. Insertion of leading zeroes is not required to use PIC numbers.

Provide PIC Information

[0258] After assignment of a PIC to a DDA number, the PIC database record is populated with data provided by the requesting bank. The data elements for a PIC record include: company name, company address, bank name, routing number, account number, account type, status, etc.

[0259] Account type, as defined by the system indicates whether the account is a business account, internal account, for correspondent relationships, or a consumer account for use in a home banking context. The PIC database record also indicates the EPN/CHIPS enabled status of an account if applicable. This status is important

as some banks or accounts may not be registered with both EPN and CHIPS. The status field reads as enabled, disabled or suspended. If a bank is both EPN and CHIPS registered, PIC numbers are enabled for both payment systems at the time of PIC assignment. In addition to system specific status, the PIC database also contains information related to the overall status of the PIC. This status can be active, suspended or closed. Upon assignment of a PIC, the overall status defaults to active.

[0260] Many banks rely on correspondent relationships to execute CHIPS payments. There can be multiple bank process links associated with a single CHIPS payment. These links create a CHIPS “payment chain.” As a consequence, the PIC database record, for a CHIPS enabled customer whose bank is not a direct CHIPS participant, requires an additional data element. This element contains the PIC associated with the account at the CHIPS participating bank. By capturing the PIC of the next bank account in the chain, iC&S allows for easy maintenance of correspondent relationship information. A correspondent relationship change requires a change to a single PIC in the payment chain.

Process Payments

Originate Payments

[0261] The use of PICs by buyers and sellers requires minimal changes, if any, to originating bank’s and receiving bank’s systems. The system routing number and PIC are used in the same manner as routing numbers and account numbers are used currently in, for example, EPN. PIC numbers are left justified with any leading zeros being insignificant. EPN and CHIPS recognize PIC transactions through a PIC-specific routing number contained within the payment instruction. For EPN, the flag is a nine digit system routing number. This is a unique number identifying a particular financial institution and which is assigned by, and registered with, an independent organization, preferably with Thompson Financial. For CHIPS, the flag is a four-digit number determined by the Clearing House. Once a transaction

is identified as an a system PIC transaction, the EPN/CHIPS platform scans the account number field in the payment instruction and reads the PIC to retrieve the associated account information. For all system transactions, the account number/PIC field is mandatory.

Receive Payment Instructions

[0262] Processing a PIC payment refers to the process of receiving a payment instruction, identifying the instruction as a PIC transaction, validating the PIC, translating the PIC and forwarding the payment to the beneficiary's bank. The PIC payment process of the present invention may be integrated with the processes of EPN and CHIPS. While it may be preferable initially to use the system to facilitate credit origination only, blocking all debit originations, it is possible, and may be advantageous, to use PIC for certain types of EPN debit transactions. PIC credit payments require the receiving customer to have a PIC. The payment originator is not required to have a PIC.

[0263] When an originating bank is not an EPN participant, ACH payments are routed to the Federal Reserve. The Federal Reserve recognizes PIC transactions from the system routing number and credits a system-specific settlement account. EPN then performs the PIC translation, payment delivery and settlement to complete the transaction. A similar process is used for processing CHIPS transactions in cases where an originating bank is not a CHIPS participant and sends wire transactions (with PICs) through the Federal Reserve for processing.

[0264] To execute CHIPS payments, participating banks must process payments through a CHIPS registered participant. All CHIPS registered participants and their correspondent banks are required to have PICs. Finally, implementation of the PIC concept does not adversely impact overall EPN/CHIPS platform performance.

Validate PIC

[0265] PIC numbers are validated before account information look-up can occur and before the payment can be sent to the receiver. The EPN and CHIPS platforms each have a copy of the PIC database in order to validate and translate PICs. These PIC databases reside on the EPN and CHIPS platforms and are updated daily by the master PIC database (immediately after previous day changes have been recorded). When validating the PIC against the PIC database, the payment systems return (EPN) or reject (CHIPS) the payment for the following reasons: PIC is invalid; and PIC status is not active.

[0266] EPN and CHIPS capture the return/rejection error reason, which is accessible by internal customer service. If a PIC is rejected in CHIPS, it is rejected with the generic “invalid” response currently used by the system.

Translate PIC

[0267] Once a PIC is validated, EPN/CHIPS translates the system routing number and PIC to the bank routing number and customer DDA number using information contained in the PIC database. Because of the direct linkage between PIC and EPN/CHIPS, PIC translation is available during all EPN/CHIPS processing windows. For CHIPS PIC transactions, where a chain of banks is involved, CHIPS retrieves and translates the PIC of all banks in the payment chain and places the information in the payment record. The system may truncate certain fields. The PIC numbers are included in the outbound transaction.

Handle Rejected/Returned Instructions

[0268] ACH reversals are allowed for PIC transactions. For EPN returns from receiving banks, EPN identifies which returns are PIC transactions and require reverse translation. EPN then translates the bank routing number and DDA number back to the system routing number and PIC prior to returning a transaction to an originating bank. Furthermore, the system passes all returns to originating

banks with the same information and in the same format that the original instruction was received. However, a system tag is added to identify returns. A generic reason code (i.e., invalid account) and addenda records are also included in all returned transactions.

[0269] Figs. 2 through 4 provide a summary description of the payment processes.

Provide Customer Service

[0270] The system provides customer service to banks through the current EPN and CHIPS customer service organization(s). Preferably, the system provides service directly to banks, and their correspondents while the responsibility to communicate directly with business customers resides with banks.

Provide Service Channels

[0271] PIC customer service is provided to banks via multiple channels to accommodate individual bank preferences/processes. PIC service is available 24 hours per day, seven days a week via a combination of the following service channels and mediums: Self-service web-site; Bulk-file transfer; Messaging; Email; and Telephone.

[0272] The service website is accessible via the SWIFTNet private network. In the future, banks may access the website via the Internet. The system website includes functionality related to frequently asked questions, on-line help and contacts. Customer service email addresses and telephone numbers are listed on the web site.

Look-Up PIC

[0273] A PIC look/up search capability is provided to banks via multiple customer service channels. The PIC lookup provides a method for bank (or approved service provider) users to view the PIC database to retrieve their PICs and related information. The PIC lookup function is searchable on the following fields in the

PIC data record: PIC, DDA number, account name and address. A public PIC directory enables buyers to find seller PIC numbers. Bank customers are able to opt-in to the public PIC directory.

Maintain PIC

[0274] Maintaining accurate account information is critical to achieving a high incidence of PIC transactions. For this reason, the system philosophy related to PIC service places responsibility for PIC database maintenance with participating banks. Only authorized system customer service, participating bank or approved service provider personnel can access the PIC database to perform maintenance functions. When viewing or maintaining the PIC database, users are only granted access to PICs that relate to accounts of their bank customers. Basic PIC maintenance functions include: Close PIC; Change PIC Database Information; Transfer PIC; and Administer Profiles.

[0275] Close PIC. When a business customer closes a PIC through its bank, the PIC is retired forever. While PIC closings are infrequent, they may occur in cases of account closings due to bankruptcy or merger.

[0276] Change PIC Database Information. Changes to PIC database information are file transfer-based, message-based, web-based or telephone-based. With regard to file transfers, the system accepts PIC maintenance request files several times a day, in both XML and flat file formats. XML is the preferred file format. Files follow a format determined by the system and must be authorized prior to transfer.

[0277] All information contained in the PIC database record (except for the PIC) can be changed through maintenance requests. However, requests made through the website or through messaging are restricted to changes of non-transaction critical information. Information considered transaction critical is any information required to process a payment transaction including routing and DDA numbers.

[0278] All maintenance requests include an effective date field. Once a request is received by the system the changes are not applied until the effective date

specified. If no effective date is identified, the changes appear immediately and are placed in production systems the following day. Furthermore, the system provides banks with the option to require approval for all web or message-based maintenance requests of critical information.

[0279] Only a single change request and a single transfer request can be pending for a given PIC at any time. The system processes both if the transfer effective date is after the change effective date. Otherwise, the change request is ignored at the time of transfer. If a user needs to make additional changes after submitting a maintenance request, they must cancel the original request and include all changes in a new request. Bank users have the ability to access all outstanding and previously applied updates to its PICs.

[0280] Also, when web or message-based maintenance requests are submitted, initiators of the request receive an automated confirmation screen or message that contains relevant transaction advice.

[0281] Finally, participating banks have a copy of the PIC database for their DDA accounts to facilitate look-up and assist in identification of on-us transactions. To update the bank's customer PIC database copy, a bank requests an update file. This request includes a specific date range to determine the update requirements. If no date range is provided, the system default is to provide all updates since the date of the requesting bank's last recorded update request. Updated PIC files are returned to banks in the same format in which they were received. As necessary, the system provides banks with a complete file of all PIC data belonging to their customers.

[0282] Transfer PIC. The transfer process allows a PIC to remain with a business customer regardless of changes in the business customer's bank relationship. To complete a PIC transfer between banks, the receiving bank, or its approved service provider, must initiate the transfer. Initiation of a transfer requires proper authorization from the business customer associated with the transferring PIC. The receiving bank must also provide the system with the transferring bank and the new

bank account information for validation purposes. Once a transfer is requested, it generates a notification to the bank surrendering responsibility for the PIC.

[0283] Administer Profiles. As part of the system, direct or indirect participants of the EPN and CHIPS systems must establish bank entity profiles and individual user profiles. The self-serve website has an administrator restricted section that provides functionality to administer both types of profiles. The system requires four super administrators (two for EPN and two for CHIPS) from each participating bank. Super administrators are responsible for managing their bank's user profile information. Critical data elements in the bank profile are restricted, including settlement, routing, translation and billing information. Changes to critical information occur through off-line processes. The functions to administer user profiles include changing access controls, resetting password, creating user profiles and displaying bank user profiles records. Administrative updates occur in real-time.

Provide Technical and Bank Support

[0284] The system provides bank support via the current EPN and CHIPS customer service organization. System administrators and customer service representatives have user profiles to control access and restrict functionality and administrative rights. Customer service is able to perform all maintenance functions on behalf of banks and respond to both phone and email inquiries.

Manage Risk and Control System

[0285] To ensure proper security of PIC data and to protect against unauthorized changes of information, the system has a methodology for system control and risk management. There are four key areas of this methodology:

- Manage technical architecture
- Authenticate and control user access
- Track PIC database activity

- Secure information

[0286] Technical architecture is outside the scope of the business requirements effort and is not outlined in this document.

Authenticate and Control User Access

[0287] Participating banks are responsible for designating four super administrators, two for the EPN platform and two for the CHIPS platform. Super administrators are responsible for PIC account maintenance. Each super administrator can delegate responsibility for PIC maintenance and administration as they see fit; however, only authenticated users are permitted to make changes to the system database. An authenticated user must have a system user-profile. This profile contains several data elements including: user name, password, title, bank name, contact information, etc. There is also an employee type data element that identifies whether a user is bank or a system employee. User profiles are used to authorize a user's access, maintenance and administrative rights via all channels of communication.

[0288] Access to the system website is restricted by a login that requires a user to enter a username and password. User names and passwords must be at least six characters long, be a combination of alpha and number characters, and changed on a periodic basis. The user profile contains a status field that indicates if the profile is active, disabled or closed. A disabled status occurs when a profile has not been accessed for one month or after three failed login attempts. Once a profile is disabled, the user is restricted from logging onto the website until an administrator resets the user profile status to active. Changes related to all non-transaction critical PIC data such as name and address are allowed via the website. However, changes to transaction critical information (e.g., routing and account number) are handled off-line. This limitation allows the Clearing House to more closely monitor changes that affect PIC transaction processing.

Track PIC Database Activity

[0289] To manage risk, all modifications to the PIC database are tracked and logged. Changes to the PIC database, such as PIC assignment, maintenance and transfer are captured in a PIC activity log including the “before and after” information related to modified fields, the name of the user who performed the modification, and the date and time the change occurred. The activity log is accessible to bank super administrators via the website. All other payment related activity is tracked on the current EPN/CHIPS platform and is not stored in the PIC database.

Secure Information

[0290] Bank customer DDA information is confidential and cannot be used for anything other than PIC services initiated by participating banks on behalf of their customers. Sensitive data on the PIC master and EPN/CHIPS databases is encrypted. At a minimum, routing numbers, DDA numbers and taxpayer identification numbers must be encrypted. To bolster security, the system preferably uses of SWIFTNet to receive and transmit customer enrollment, update and information files between the system and participating banks. For SWIFTNet and ConnectDirect transmissions, the system relies on security inherent to the network and software, as well as smart cards/digital certificates to receive and transmit customer enrollment, update, and information files. Furthermore, SWIFTNet has closed user groups that restrict unknown entities from accessing the network. In the future, the Internet may be used for transmission. In this instance, the system requires a minimum of 128-bit or Triple Des encryption.

Bill Participants

[0291] Standard pricing for PIC transactions is established. PIC related transaction fees are tracked in the current EPN/CHIPS platform and do not require a separate billing advice. Rather, EPN/CHIPS bills include the number of PIC

transactions performed during the billing period and related charges. Banks make appropriate payment to the Clearing House for PIC services electronically.

Provide Reports

[0292] Because the PIC feature is an enhancement to the existing Clearing House payment systems, there are no additional reporting requirements related to the PIC feature. All payments related and statistical reports are provided through existing EPN and CHIPS reporting capabilities.

[0293] Reporting requirements related to the PIC database and maintenance functions are handled through bulk file transfer and the self-serve website. Banks can receive a file that includes all customer PICs. Reporting requirements for maintenance functionality include pending/completed PIC maintenance requests and listings of user profile usernames and passwords. This information is accessible via the iC&S website.

[0294] Several processes with respect to the system are described below with reference to Figures 8 through 33. Interaction with the PIC system is accomplished through various screens. The following describes some of the PIC screen definitions. These screen definitions may be developed to show various processes performed using the PIC database.

PIC Screen Definitions:

Welcome Screen

[0295] Initial screen presented at IC&S web site. Contains text describing iC&S and anything else a business group wants. The Welcome Screen also contains a Login Button, user and password fields and skip Login Screen Button.

Login Screen

[0296] Contains entry fields for Usercode and Password. Successful login brings you to Menu screen. After a User logs in we know which bank the user belongs to and his/her access rights.

Menu screen

[0297] Contains initial access to all capabilities that is defined for the user. Those things not allowed for the user are not visible or accessible. Those users that have Update capability will also have Inquiry capability by default. Possible Options are

| | | | |
|-----------------|---------|---|----------------------------|
| PIC | Inquiry | - | Update |
| PIC Activity | Inquiry | - | Update (approve/cancel) |
| Member | Inquiry | - | Update |
| Member Activity | Inquiry | | |
| User | Inquiry | - | Update |
| User Activity | Inquiry | | |

PIC Inquiry

Inquiry by PIC

[0298] This inquiry will display the PIC Detailed Screen. The PIC does not have to belong to the controlling bank. Do we show all account information? If inquiry is done to re-route payments the account information will be needed.

Inquiry by Account number

[0299] This inquiry is restricted to the PICS owned by the banks. Returned is the PIC List Screen. If there is a list then the user could select an entry for the detailed information.

Inquiry by Name

[0300] This inquiry is restricted to the PICS owned by the banks. Returned is the PIC List Screen. If there is a list then the user could select an entry for the detailed information. The inquiry should be able to be refined by city and/or state.

PIC List Screen

[0301] This screen will list possible candidates found from an inquiry. It will display PIC, RT number and Name and Address info, (as much as possible).

PIC Detail Screen

[0302] This screen will show detailed information regarding a specific PIC. If the user is from not from the controlling bank customer contact information is not displayed. If the user is not allowed to see account information the account number is not displayed.

PIC Activity Inquiry.

[0303] This permits inquiry by “closed/open” activity, and includes the option of looking at all closed activity. A qualifier by date range and/or PIC number or account number is used. This functionality Also allow inquiry by RT number. The results of the inquiry returns the Activity List screen.

PIC Activity List

[0304] This screen list activity records based on the Activity Inquiry Screen. It will summarize the status, action, for an PIC activity. Selecting an activity record will display the PIC Activity Detail screen.

PIC Activity Detail

[0305] This screen will contain all of the detail information regarding the activity record. If the activity record is open and the USER has update capabilities the activity record may be approved or cancelled. If the bank reviewing has 2-step approval process the activity goes in as proposed and must be approved by someone with PIC update capabilities. If the bank has dual operator approval option then the approval operator must be different from the entering operator.

Member Inquiry

[0306] This screen allows an authorized user to view MEMBER profile information. This screen will display bank names, account, wire chaining and other configuration information. If a bank list exists it is displayed and if categories are defined for the bank they are also displayed.

Member Update

[0307] This screen allows changes to the MEMBER profile by an authorized user. An add will be performed by IC&S staff after signup information is gathered. Changes are made by the Member. A delete may only be performed by IC&S staff. When a change is made the member profile is immediately updated and a closed member activity record is automatically created.

Member Activity Inquiry

[0308] This screen is used to display the Member Activity List screen by an authorized user. The activity may be selected by date range.

Member Activity List

[0309] This screen is used to display a list of member activity update records. It will display a summary of activity and dates. Selecting an activity record will display the Member Activity Detail screen.

Member Activity Detail Screen

[0310] This screen is used to display the Member profile change detail information.

USER Inquiry

[0311] This screen allows an authorized user to view USER profile information. This screen will display all access capabilities of a USER.

USER Update

[0312] This screen allows changes to the USER profile by an authorized user. When a change is made the USER profile is immediately updated and a closed USER activity record is automatically created.

USER Activity Inquiry

[0313] This screen is used to display the USER Activity List screen. The activity may be selected by date range.

USER Activity List

[0314] This screen is used to display a list of USER activity update records. It will display a summary of activity and dates. Selecting an activity record will display the USER Activity Detail screen.

USER Activity Detail Screen

[0315] This screen is used to display the USER profile change detail information.

USER OPTIONS MATRIX

| | | | |
|----------------|---------|---------------------------------|---------------------|
| PIC | Inquiry | Update | Account Information |
| PIC Activity | Inquiry | Update | |
| Member | Inquiry | Update | |
| Member Account | Inquiry | (If Member Update then Inquiry) | |
| USER | Inquiry | Update | |
| USER Account | Inquiry | (If USER Update then Inquiry) | |

PIC Maintenance

[0316] The maintenance of PIC fall within 4 categories, Add, Change, Delete(close), and Transfer.

[0317] PIC ADD: This screen is used to create a new PIC. Some of the fields are required while others are optional. Required fields are the RT number and DDA number that represent the account, Name and Address of the customer and the CHIPS/EPN enabled flag. Optional fields are the Contact information. If CHIPS enabled an optional "wire" name and address may be entered. If the "wire" name and address is not specified the customer name and address will be truncated to fit into this name and address.

[0318] PIC CHANGE: This screen is used to change an existing PIC. It will contain all the fields that an ADD screen contains. Any field may be changed except the RT number field. This may be changed only with a PIC Transfer screen.

[0319] PIC TRANSFER: This screen is used to move a PIC from one bank to another bank. The new bank must enter the new RT and DDA number for the account, and also provide the old RT and DDA number. Any of the other fields

may also be changed at this time. The “old” bank will be notified of the transfer and have 48 hours to contest the transfer. All transfer will be effective 2 days after approval.

[0320] PIC DELETE: PICS are never actually deleted, but are considered to be closed. They can be reopened at a later date.

System Processes

[0321] Figures 8 and 9 together show a Create a Bank Profile process. This process is accomplished mostly offline by internal users at a depository institution. In Figure 8, at step S1000, the user is presented with their home page. At step S1002, the user clicks on an icon for Create a Bank Profile. The user is then presented, at step S1004 with a blank bank profile page. At step S1006, the user enters data and at step S1008 hits submit. At step S1010, if the bank is in the system, the flow proceeds to step S1012, at which point the user is presented with an error page saying that the bank has already been added. If the bank is not in the system, then at step S1014, it is determined if the information has been entered correctly. If so, the flow proceeds to section A2 of Figure 9.

[0322] At Figure 9, at step S1016, the user is presented with a screen that has the information they typed in and the user is asked to either create a bank profile or change entered data. At step S1018 it is determined whether the create bank profile or the don't create bank profile has been pressed. If the create a bank profile button is pressed at step S1020, then flow proceeds to step S1024 at which the user is presented with a success screen that the profile has been created and the process terminates. If the don't create a profile button is pressed at step S1022, then flow proceeds to step S1026 at which the user is presented with the request cancelled screen and the process terminates.

[0323] Figure 10 is a flowchart of a Delete a Bank Profile process, accomplished by users with proper authority and access at the trusted third party. At step S2000, the user is presented with their home page. At step S2002, the user clicks on delete

a bank profile. At step S2004, the user is presented with a page offering the choice of entering a bank name or selecting from a listed bank name. If the user clicks on a bank name, at step S2005, the flow proceeds to section A3 of the figure. If the user types in a bank name at step S2006 and clicks on submit at step S2007, then at step S2008 if the bank exists on the system, the flow proceeds to section A3. Otherwise, flow returns, with an error message, to step S2004, and the user is again asked to enter a bank name or select a bank name.

[0324] At step S2010 the user is presented with a bank profile for the selected bank. Then, to delete the bank profile, at step S2012 the user clicks on delete. At step S2014 the user is presented with a screen asking if they are sure they want to delete the profile. If at step S2016 the user clicks at step S2017 on delete, then at step S2018, the user is presented with a Success screen and the flow terminates. On the other hand, if the user at step S2019 clicks on Do Not Delete, then at step S2020, the user is presented with a Request Cancel screen and the flow terminates.

[0325] Figure 11 shows the flow for a Modifying a Bank Profile process. This is a permission based process, whereby only those users with proper permission may perform this process. At step S2100, the user is presented with their homepage. At step S2102, the user clicks on Modify a Bank profile. If it is determined at step S2104, that the user is not a trusted third party (TTP) user, then at step S2106, the user is presented with an editable version of the bank profile. At step S2108, the user enters changes, then at step S2110, the user clicks on submit. If it is determined at step S2112 that the required fields are completed correctly, then the flow goes to section A4. If it is determined at step S2104, that the user is a trusted third party (TTP) user, then at step S2114, the user is presented with a page and asked to enter a bank name or click on a name. If at step S2114 the user sees the bank name, then at step S2116, the user clicks on the bank name and flow proceeds to section B4. If the user is not provided a bank name to click, then at step S2118, the user types in a bank name. At step S2120, the user clicks on submit, then at step S2122, the user is asked whether their bank exists in the system. If yes, flow

proceeds to section B4. If no, flow proceeds back to S2114 with an error message added to what is presented to the user. At step S2124, the user is presented with a screen that has the modification of the bank profile and asked if they want to modify or cancel the modification request. At step S2126, the user must decide to either modify or not to modify. If the user decides to modify at step S2126, then at step S2128, the user clicks on modify, and at step S2130, the user is presented with a success screen and the flow ends. If the user decides not to modify at step S2126, then at step S2132, the user clicks on do not modify. At step S2134, the user is presented with a cancel request screen stating that the user was not modified and the flow ends.

[0326] Figure 12 shows the flow for a Change a Route process. This process is accomplished by the trusted third part. At step S2200, the user is presented with their homepage. At step S2202, the user clicks on change a route. At step S2204, the user is presented with a page to enter the 9 digit bank number or bank name. At step S2206, the user is presented with a page and asked for a route number and an effective date. At step S2207, the user clicks on View Routes. At step S2208, the user is presented the route table for that bank and the flow proceeds to section A5. At step S2209, the user enters data and the flow proceeds to section A5. At step S2210, the user must choose: Add Route, Move Route, Delete Route, View Route Table. At step S2212, the user has clicked on Add Route, and the flow proceeds to step S2214, to determine whether Route exists. If no, flow proceeds to section B5. If yes, flow proceeds to step S2216, where it is determined if required fields has been completed correctly? If no, flow proceeds, with an error message, to section B5. If yes, then at step S2218, the user is presented with a screen that asks if they are sure they want to do that activity. At step S2220, the user must decide to do activity or cancel request for activity. At step S2222, the user has clicked on "do activity." Then at step S2224, the user is presented with a Success screen and the flow terminates. At step S2226, the user has clicked on "cancel request for activity." Then at step S2228, the user is presented with a

Success screen stating the request was canceled and the flow terminates. At step S2230, the user has clicked on Delete Route, then at step S2232, it is determined if Route exists. If yes, flow proceeds to step S2216. If no, flow proceeds to section B5. At step S2234, the user clicks on Move Route which is a placeholder. At S2236, the user clicks on View Route Table, and the flow proceeds to section B5.1.

[0327] Figure 13 is a flowchart of the View PIC Activity log process. In this process, all user related activities should appear based upon a search criteria. At step S2400, the user is presented with their homepage. At step S2402, the user clicks on the View PIC Activity Link. At step S2404, it is determined if the user has access to multiple banks. If yes, then at step S2406, the user is presented with page to enter search criteria for activity and a list of the most recent activities for all banks they have access to. At step S2408, the user is presented with a choice to enter search data or click on an activity. At step S2410, user that has entered search criteria clicks on the Submit Button. If an activity is selected, the flow proceeds to section C6. If at step S2404 the answer is no, then at step S2411, the user is presented with page to enter search criteria for activity and a list of the most recent activities for all PICs which he has access to and then the flow proceeds to step S2408. At step S2412, it is determined if the user requested find selected criteria? If no, then flow proceeds to section A6 with an error message. If yes, then flow proceeds to step S2414, where it is determined if the user has access to this information. If no, then flow proceeds to section A6, with an error message. If yes, then flow proceeds to step S2416, the user is presented with search results, 20 per page. At step S2418, the user clicks on the PIC activity to View, and at step S2419, the user is presented with page of activity information and the process terminates. Alternatively, at step S2420, the user clicks to search again, and the flow proceeds to section A6.

[0328] Figure 14 is a flowchart of the View Bank Profile Activity log process. At step S2500, the user is presented with their homepage. At step S2502, the user clicks on the View Bank Profile Activity Log Link. At step S2504, it is determined

if the user has access to multiple banks. If yes, then at step S2506, the user is presented with Bank Profile activity for the multiple banks they have access to. If the activity is in the future and the user has right to perform that activity, a CANCEL button preferably appears to allow for the cancellation of that pending activity. At step S2508, it is determined if the user found what they wanted on log. If yes, then the process terminates. If no, then the flow proceeds to step S2510, where the user enters Search Criteria. At step S2512, the user clicks on Find an Activity and the flow proceeds to section A7. If at step S2504, the answer is no, then at step S2514, the user is presented with the Bank profile activity log after which the flow proceeds to step S2514.

[0329] Figures 15A and 15B are a flowchart of the Create a User process. At step S2600, the user is presented with their homepage. At step S2602, the user clicks on the Add A User Link. At step S2604, it is determined if the user has access to create the users at Multiple banks. If yes, then at step S2606, the user is presented with a list of banks they have access to add the users to. At step S2608, the user hits a bank name. At step S2610, the user gets page with the types of the users they are allowed to create. If at step S2604 the answer is no, then flow proceeds directly to step S2610. At step S2612, the user clicks on a the user type. At step S2614, the user is presented with an a user profile setup screen. At step S2616, the user enters requested information. At step S2618, the user hits submit. At step S2620, it is determined if the required fields were completed correctly. If yes, flow proceeds to section A8. If no, flow proceeds to step S2614 with an error message. At step S2622, the user is presented with a screen that has the information they typed in on it and asked to either Create User or Change Data. At step S2624, the user has a decision whether to hit Create User, Don't Create User, or hit Back Button in browser. If at step S2625 the user hits Create User, then at step S2626, the user is presented with a Success screen that the user has been created and the process terminates. If instead the user, at step S2627 hits Don't Create User, then flow proceeds to step S2628, and the user is presented with Request Cancelled

screen and process terminates. If the user, at step S2629 hits the Back Button in the browser Window, then at step S2630, the user may change data and the flow proceeds to section D8.

[0330] Figures 16A and 16B are a flowchart of the Delete a User process. Note that a user cannot delete themselves. At step S2700, the user is presented with their homepage. At step S2702, the user clicks on the Delete User Link. At step S2704, it is determined if the user has access to delete the users from Multiple banks. If yes, then at step S2706, the user is presented with a list of banks they have access to delete the users from. At step S2708, the user hits a bank name. At step S2710, the user is presented with a page with the names of the users they have access to delete. There are check boxes next to each name. If at step S2704 the determination is no, then the flow proceeds directly from step S2704 to step S2710. At step S2712, the user can either click name or enter a name or UserID in a provided form field to search for that user. If at step S2714, the user clicks on the Check box next to the user's name, then at step S2715, the user hits Delete and the flow goes to section A9. If instead, at step S2716, the user enters name or UserID in form field, then at step S2717, the user hits search for user, and at step S2718, the user is presented with a page with a user or users that match that search and the flow proceeds to section B9. At step S2720, the user is asked whether they are sure they want to delete user. At step S2722, the user decides whether to Delete Profile or Don't Delete Profile. If the user at step S2724, hits Delete, then at step S2725, the user account is marked for deletion in the system database and at step S2726, the user is presented with a Success screen that user has been deleted, and the process terminates. If instead, at step S2727, the user selects Don't Delete Profile, then at step S2728, the user is presented with a screen stating their Request to delete has been cancelled and the process terminates.

[0331] Figures 17A and 17B are a flowchart of a Modify a User process. At step S2800, the user is presented with their homepage. This is a permission based process, whereby only those users with proper permission may perform this

process. At step S2802, the use clicks on the Modify User. At step S2804, it is determined if the user has access to modify users from multiple banks. If yes, then at step S2806, the user is presented with a list of banks they have access to modify users from. At step S2808, the user hits a bank name. Then at step S2810 the user is presented with a page with the names of users they have access to modify. If at step S2804 the answer is no, the flow proceeds directly from step S2804 to step S2810. At step S2812, the user can either click a name or enter a name in a provided form field to search for that user. If, at step S2813, the user clicks a name, then flow proceeds to section A10. If instead, at step S2814, the users enters a UserID in the form field, then at step S2815, the user hits search for user, and at step S2816, the user is presented with a page with a user or users that match that search and the flow proceeds to section B10. At step S2818, the user is presented with an editable version of that person's profile. At step S2820, the user modifies data. At step S2822, the user hits Modify. At step S2824, it is determined if the data was entered correctly. If, no, flow goes back to step S2818 with an error message. If yes, then at step S2826, the users are asked are they sure they want to modify a user or users. At step S2828, the user decides whether to hit Modify or Don't Modify User. If, at step S2829, the user hits Modify, then at step S2830, the user account marked modified in the PIC database, and at step S2831, the user is presented with a Success screen that user has been modified and process terminates. If instead, at step S2832, the user hits Don't Modify, then at step S2833, the user is presented with a request Cancel screen and the process terminates.

[0332] Figure 18 is a flowchart of a View a User Profile process. At step S2900, the user is presented with their homepage. This is a permission based process, whereby only those users with proper permission may perform this process. At step S2902, the user clicks on View User Profile. At step S2904, it is determined if the user can View bank users from multiple banks. If yes, then at step S2906, the user is presented with the list of banks they have access to view user profiles at or

can search for bank, and at step S2908, the user clicks on Bank Name. At step S2910, the user is presented with a page with the names of users they have access to view a profile for. If the user can only view their own profile, the process skips to section C11. If the determination at step S2904 is no, the flow proceeds directly to step S2910. At step S2912, the user can either click on a name or enter a name or ID in a provided form field to search for that user. If, at step S2913, the user Clicks On Name, then at step S2914, the user is presented with the User Profile and the process terminates. If instead, at step S2915, the user enters name in the form field, then at step S2916, the user hits search for user, and at step S2917, the user is presented with a page with a user or users that match that search and the flow proceeds to section B11. If, at step S2918, the user enters User ID in the form field, then at step S2919, the user hits search for user, and at step S2920, the user is presented with the User Profile and the process terminates.

[0333] Figure 19 is a flowchart of a Log On to The PIC System process. This is a permission based process, whereby only those users with proper permission may perform this process. At step S3000, the user Logs on to SWIFTNet. At step S3002, the user is presented with a browser screen prompting for their userid and password. At step S3004, the user enters Data. At step S3006, the user clicks on Submit to submit the data. At step S3008, it is determined if the user is authenticated and authorized. If no, flow returns to step S3002 with an error message. If yes, at step S3010, it is determined if the required fields have been completed correctly. If no, flow returns to step S3002 with an error message. If yes, then at step S3012, the user is presented with their homepage and the process terminates.

[0334] Figure 20 is a flowchart of a Change a Password process. First time new users accessing the system are requested to change the password. Also, request to change password is automatic on password expiration. At step S3100, the user is presented with their homepage. At step S3102, the user clicks on a User Change Password link. At step S3104, the user is presented with a page to enter their old

password and new Password, preferably two times. At step S3106, the user clicks Change Password. At step S3108, it is determined if the password has been changed. If yes, at step S3110, the user is presented with a Success screen and the process terminates. If no, an error message is presented and the process terminates.

[0335] Figure 21 is a flowchart of a Reset User process. At step S3200, the user is presented with their homepage. At step S3204, the user clicks on User Reset User Link. At step S3206, the user is presented with a page including a list of users as well as the option to enter username for search. At step S3208, the user chooses whether to enter search for user or to click on a user name. If a search is to be performed, at step S3209, the user enters a user name and clicks on the Submit Button and flow proceeds to section A14. If the user wishes to click on a displayed user, then at step S3210, the user clicks on a user's name and the flow proceeds to section A14. At step S3211, the user has the choice to reset the user. At step S3212, the user clicks Reset User. At step S3214, the user is presented with message to verify Reset/Don't Reset. At step S3216, the user will verify to reset the user. At step S3217, if reset selected the user is presented with Success screen and the process terminates. If reset is not selected, at step S3218, the user is presented with a request cancelled screen and the process terminates. If instead, at step S3219, the user clicks Don't Reset, the flow also proceeds to step S3219.

[0336] Figures 22A and 22B are a flowchart of Transfer a PIC process. This is a permission based process, whereby only those users with proper permission may perform this process. At step S3300, the user is presented with their homepage. At step S3302, the user clicks on the Transfer a PIC link. At step S3304, the user is presented with a page where he can enter a PIC number to transfer. At step S3306, the user enters the PIC. At step S3308, the user clicks on the Submit Button. At step S3310, it is determined if the PIC exists in the system. If no, the flow proceeds to section B15. If yes, at step S3312, it is determined if the routing number belongs to acquiring bank. If no, flow proceeds to section A15. If yes, and the PIC already belongs to an acquiring bank, flow proceeds to section B15. At

step S3314, the user is presented with a PIC transfer screen. At step S3316, the user enters Current requested PIC data. At step S3318, the user clicks on the Submit Button. At step S3320, it is determined if the current PIC info is complete and valid. If no, flow returns to step S3314. If yes, at step S3322, the user is presented with a PIC Customer Information screen and asked to confirm and Request Transfer. At step S3324, it is determined if the customer information is Correct. If no, flow returns to step S3314. If yes, flow proceeds to section C15. At step S3326, the user is presented with a screen for them to enter new bank information. At step S3328, the user enters PIC bank information and an effective date. At step S3330, the user clicks on the Submit Transfer Button.

[0337] At step S3332, it is determined if the PIC transfer info is complete and verified. If no, flow returns to step S3326. If yes, at step S3334, the user is presented with a screen for them to verify new PIC information. At step S3336, it is verified whether the PIC transfer information is correct and the decision is made to transfer/don't transfer. If transfer is selected, flow proceeds to section D15. If don't transfer is selected, the process terminates. At step S3338, a PIC transfer is entered into a database. At step S3339, the user is presented with a success screen and an explanation of the next steps. At step S3340, the releasing bank is notified of a pending PIC transfer.

[0338] Figure 23 is a flowchart of a Create a PIC process. At step S3400, the user is presented with their homepage. At step S3402, the user clicks on Create A PIC. At step S3404, a 7.0 NYCH/user validation Routine is performed. At step S3406, the user is presented with the create a PIC screen. At step S3408, the user fills out all the required information. At step S3410, the user clicks on Submit and the flow proceeds to section A16. At step S3412, the user is presented with the create PIC screen with data previously entered and may change data and the flow proceeds to step S3408. At step S3414, it is determined if the required fields are completed correctly. If no, flow proceeds to section B16 with an error message. If yes, at step S3416, it is determined if the DDA number does not have a PIC. If no, that is, if

the DDA already has a PIC, the flow proceeds to B16 with an error message. If yes, at step S3418, the user is presented with a screen that has the information they entered and asked to verify - Create PIC, or the user can hit the back Button in the browser window. At step S3420, the user decides whether to hit create PIC or hit the back Button in the browser window. If at step S3421, the user hits Create PIC, then at step S3422, the user is presented with a Success screen that user has been deleted and the process terminates. If instead, at step S3423, the user hits back button in browser, the flow reverts to section B16.

[0339] Figures 24A and 24B are a flowchart of a Close a PIC process. PICs are never deleted, merely closed. At step S3500, the user is presented with their homepage. At step S3502, the user clicks on Close PIC. At step S3504, the user is presented with a page and asked to enter a PIC Number. At step S3506, the user types in a PIC number. At step S3508, the user clicks on submit. At step S3510, it is determined if the PIC exists in the system. If no, flow returns to step S3504 with an error message. If yes, at step S3512, it is determined if the user has a right to Close that PIC. If no, flow returns to step S3504 with an error message. If yes, flow proceeds to section A17. At step S3514, it is determined if the PIC has activity pending. If yes, at step S3516, the user is presented with the current PIC data, which contains an ALERT with an activity pending message. At step S3518, the user clicks to view activity pending data. At step S3520, the user is presented with Activity Screen. At step S3522, it is determined if the User is the owner of the Pending Activity. If no, the process terminates. If yes, at step S3524, the user types in date and comments. At step S3526, the user clicks on Close PIC and flow proceeds to section B17. If the answer determined at step S3514 is no, then at step S3528, the user is presented with the PIC screen with space for effective date and comments. At step S3530, the user types in dates and comments. At step S3532, the user clicks on Close PIC and the flow proceeds to section B17. At step S3534, it is determined if the date is a valid date. If no, the flow proceeds to section C17 with an error message. If yes, at step S3536, the user is presented with a screen

asking them are they sure they want to close PIC or Don't close PIC. At step S3538, the user decides between Close PIC and Don't Close PIC. If, at step S3539, the user hits Close PIC, then at step S3540 the user is presented with a Success screen that PIC has been closed and the process terminates. If instead, at step S3541, the user hits Don't Close PIC, then at step S3542, the user is presented with a Request Canceled Page and the process terminates.

[0340] Figures 25A and 25B are a flowchart of a Reactivate A PIC process. To reactivate a PIC, it must belong to the bank. At step S3600, the user is presented with their homepage. At step S3602, the user clicks on Reactive PIC. At step S3604, the user is presented with a page and asked to enter a PIC. At step S3606, the user types in a PIC. At step S3608, the user clicks on submit. At step S3610 it is determined if that PIC exists in the system. If no, flow reverts to step S3604 with an error message. If yes, at step S3612 it is determined if the user has rights to reactivate that PIC. If no, flow reverts to step S3604 with an error message. If yes, flow proceeds to section A18, where, at step S3614 it is determined if the PIC has activity pending. If yes, then at step S3616, the user is presented with the current PIC data which contains an ALERT with an activity pending message. At step S3618, the user clicks to view activity pending data. At S3620, the user is presented with Activity Screen. At step S3622, it is determined if the user is the owner of the Pending Activity. If no, the process terminates with an error message. If yes, at step S3624, the user types in Date and comments. At step S3626, the user clicks on Reactivate PIC and the flow proceeds to section B18. If the answer is no at step S3614, then at step S3628, the user is presented with the PIC screen with space for effective date and comments. At step S3630, the user types in Date and comments. At step S3632, the user clicks on Reactivate PIC and the flow proceeds to section B18, where at step S3634, it is determined if the account is being recycled. If yes, the process terminates with an error message. If no, then at step S3636, it is determined if the date is a valid date. If no, flow proceeds to section C18 with an error message. If yes, then at step S3638, the user is presented with a

screen asking them are they sure they want to Reactivate PIC or Don't Reactivate PIC. At step S3640, the user decides whether to select Reactivate PIC or Don't Reactivate PIC. If, at step S3641, the user clicks Reactivate PIC, then at step S3642, the user is presented with a Success screen that PIC has been Reactivated and the process terminates. If instead, at step S3643, the user clicks Don't Reactivate PIC then at step S3644, the user is presented with a Request Canceled Page, and the process terminates.

[0341] Figure 26 is a flowchart of Contest/Release A PIC Pending Transfers process. At step S3700, the user is presented with their homepage. At step S3702, the user clicks on the Transfer Pending PIC Alert(s) or contest transfer. At step S3704, the user is presented with a page explaining the Transfer(s) pending. At step S3706, the user decides whether to Contest Transfer, Release, or Do Nothing. If, at step S3707, the user clicks on Contest, the flow proceeds to section A19. If, at step S3708, the user does nothing, then at step S3709, transfer occurs after a predetermined number of days. If, at step S3710, the user clicks on Release the flow proceeds to section B19. At step S3712, the user is presented with a screen that explains the next steps. At step S3714, the user checks a box to indicate a read message. At step S3716, system puts PIC in Contested status, and the process terminates. At step S3718, the user is presented with a screen that asks if they are sure they want to release. At step S3720, it is determined if there has been a release. If release, then at step S3721, the user clicks on Release, and at step S3722, the user is presented with a success screen and the process terminates. If don't release is selected, at step S3723 the user clicks on Don't Release and at step S3724, the user is presented with a request cancelled screen, and the process terminates.

[0342] Figures 27A and 27B are a flowchart of an Approve A PIC Activity process. Examples of PIC activity that needs to be approved are Add, Delete, and Modify. At step S3800, the user is presented with their homepage. At step S3802, the user clicks on the PICS Activity to Approve on home page or PIC Approve. At

step S3804, the user is presented with a page listing the PICs they need to approve. At step S3806, the user clicks on a PIC. At step S3808, it is determined if the activity was entered by this user. If no, then at step S3810, the user is presented with a screen that shows the information for that Activity for a PIC and the process proceeds to section A20. If yes, then at step S3812, the user is presented with a screen that shows the information for that Activity for a PIC that explains they cannot approve it and the process terminates. At step S3814, the user can either Approve, Modify, Cancel. If at step S3816, the user clicks on Approve, then at step S3818, the user is asked are they sure they want to approve. At step S3820, the user decides between Approve and Don't Approve. If, at step S3821, the user hits Approve, then at step S3822, activity is marked as approved in the Ic&s database, and at step S3823, the user is presented with a success page and the process is terminated. If, at step S3824, the user selects Don't Approve, then at step S3825, the user is presented with a Success screen stating the item has not been approved and the process terminates. If, at step S3826, the user clicks on Modify, then at step S3827, the Modify A PIC process is invoked (described later). If, at step S3828, the user clicks Cancel Activity, then at step S3830, the user is asked if they are sure they want to Cancel. At step S3832, the user decides between Cancel and Don't Cancel. If, at step S3833, the user hits Cancel, then at step S3834, activity is marked as canceled in the ic&s database, and at step S3835, the user is presented cancel request screen stating the activity has been canceled, after which the process terminates. If instead, at step S3836, Don't Cancel is selected, then at step S3837, the user is presented with a Success screen stating the item has not been canceled, after which the process terminates.

[0343] Figures 28A and 28B are a flowchart of a Find A PIC process. All PICs will be retrieved based on a search criteria, including closed and pending PICs. Those PICs that a user cannot view or act upon, will also appear but be grayed out. At step S3900, the user is presented with their homepage. At step S3902, the user clicks on the Find A PIC Link. At step S3904, the user is presented with a page to

search for a PIC. At step S3906, the user makes a choice as to what they wish to search by. If, at step S3908, the user enters a DDA/RT & Number to perform the search, then at step S3910, the user clicks on the Submit Button, and at step S3912, it is determined if the required fields have been completed correctly. If no, flow proceeds to section A21. If yes, then at step S3914, it is determined if the user requested Find by Co. Name & City. If no, then flow proceeds to section C21. If yes, flow proceeds to section B21.

[0344] If, on the other hand, the search is to be by PIC number, then, at step S3916, the user enters PIC Number and the flow proceeds to step S3910. If instead the search is to be by name and address, then at step S3918, the user enters the Company Name and City and the flow proceeds to step S3910 and continues from there as discussed above. At step S3920, the user is presented with search results preferably about 20 per page. At step S3922, the user clicks on the Company PIC info to view. At step S3924, it is determined whether the User Bank is the owner of the customer. If yes, then at step S3926, the user is presented with the selected Company PIC information. If no, then at step S3928, the user is presented with the selected Company PIC information in Abbreviated form. In either event, the process is then terminated.

[0345] Figure 29 is a flowchart of View User Profile Activity Log process. At step S4000, the user is presented with their homepage. At step S4002, the user clicks on the User View Activity Link. At step S4004, it is determined if the user has access to multiple banks. If yes, then at step S4006, the user is presented with page to search for Banks and makes bank selection. At step S4008, the user is presented with page to enter search criteria for user activity and a list of the most recent user activities for which he has access to. If no, the flow proceeds directly from step S4004 to step S4008. At step S4010, the user chooses to enter search data or to click on an activity. If the user chooses to use search criteria, then at step S4012, the user enters search criteria and clicks on the Submit Button and the flow proceeds to section B22. If an activity is selected, the flow proceeds to

section C22. At step S4014, it is determined if the user request found selected criteria. If no, flow proceeds to section A22 with an error message. If yes, then at step S4016, it is determined if the user has access to this information. If no, flow proceeds to section A22 with an error message. If yes, then at step S4018, the user is presented with search results preferably about 20 per page. If at step S4020, the user clicks on the User activity to View, then step S4021, the user is presented with page of User activity information and the process terminates. Alternatively, at step S4022, the user can click to search again, and the process returns to section A22.

[0346] Figures 30A and 30B are a flowchart of a Modify A PIC process. This is a permission based process, whereby only those users with proper permission may perform this process. View Only user will not be able to perform this process. Also, anytime a user gets a data entry error, that user is returned to the screen with the data they entered pre-filled. At step S4100, the user is presented with their homepage. At step S4102, the user clicks on Modify A PIC. At step S4104, the user is presented with a page and asked to enter a PIC Number. At step S4106, the user types in a PIC number. At step S4108, the user clicks on submit. At step S4110, it is determined if that PIC exists in the system. If no, flow returns to step S4104 with an error message. If yes, then at step S4112, it is determined if the User has rights to modify that PIC. If no, flow returns to step S4104 with an error message. If yes, then flow proceeds to section A23. At step S4114, it is determined if the PIC has activity pending. If yes, then at step S4116, the user is presented with current PIC data, which contains ALERT with activity pending message. At step S4118, the user clicks to view activity pending data. At step S4120, the user is presented with Activity Screen. At step S4122, it is determined if the User is the owner of the Pending Activity. If no, an error message is produced and the process terminates. If yes, then at step S4124, the user types in Modifications, Date and comments and the flow proceeds to section B23. Step S4124 may also be entered after execution of step S4126, at which a user is presented with an activity screen.

[0347] If the answer determined at step S4114 is no, then at step S4128, the user is presented with an editable PIC Modify screen with space for effective date and the flow proceeds to step S4124. At step S4130, the user clicks on Modify PIC. At step S4132, it is determined if the Data was entered correctly. If no, then at step S4134, the user is presented with a screen that has data they typed in so it can be corrected and the flow then returns to step S4132. Once the data is determined to be entered correctly, then at step S4136, the user is presented with a screen that has data they typed in for verification and asked if they are sure they want to modify PIC and the flow proceeds to section C23. At step S4138, the user clicks on Modify PIC, and at step S4139, the user is presented with a Success screen and the process terminates. Alternatively, at step S4140, the user clicks on Cancel Modification Request, and at step S4141, the user is presented with a cancel request screen and the process terminates.

[0348] Figure 31 is a flowchart of a Perform Trusted Third Party Validation process. At step S4200, it is determined if the User can perform one or more functions on PICs for multiple banks. If no, flow returns to the calling process. If yes, at step S4202, the user is presented with a page and asked to enter a bank name. At step S4204, the user types in a bank name. At step S4206, the user clicks on submit and the flow returns to Create A PIC flowchart at step S3406.

[0349] Figure 32 is a flowchart of a Validate Bank Profile process. This is a one-time process that occurs the first time a bank profile validator logs onto the system. Before any PIC activity can be accomplished, the bank profile must be validated. At step S4300, the user enters the Userid and password, and, at step S4302, hits submit. At step S4304, it is determined if the bank profile has been validated. At step S4306, it is determined if the User is A Validator. If yes, at step S4308, the user is presented with the Bank profile page. If, at step S4310, the user approves, then at step S4312, the Bank Profile is Validated and the user is presented with a Success screen. If at step S4313, the user chooses modify, then at step S4314, the process transfers to the to "Modify A Bank Profile" flow. If no at step S4306 or if

yes at step S4304, then at step S4315, the user is presented with their personal page with appropriate menus.

[0350] Figures 33A and 33B a flowchart of an Approve A Bank Profile Modification process. This process is for future use. Bank Profile activity that would have to be approved would be, for example, Add, and Modify. The approval loop is only necessary when specified on the bank profile. At step S4400, the user is presented with their homepage. At step S4402, the user clicks on the Bank Activity to Approve. At step S4404, the user is presented with a screen that shows the information for that Bank Activity. At step S4406, it is determined if a Activity to be approved was created by user. If yes, then at step S4408, the user is presented with a screen that shows that bank activity. At step S4410, the user can either Modify, or Cancel the Request. If, at step S4412, the user clicks on Modify, then at step S4413, process Modify A Bank Profile 1.4 is called. If, on the other hand, at step S4414, the user clicks Cancel Activity, then the flow proceeds to section A. If the answer at step S4406 is no, then at step S4416, the user can either: Approve, Modify, or Cancel the Request. If at step S4418, the user clicks on Approve, then the flow proceeds to section A. If, at step S4420, the user clicks on Modify, then at step S4421, the Modify A Bank Profile 1.4. process is called. If instead, at step S4422, the user clicks Cancel Request, the flow proceeds to section A. At step S4424, the user is presented with a screen that asks if they are sure they want to do that activity. At step S4426, the user must decide to do activity or cancel request for activity. If, at step S4428, the user clicks to do activity, then at step S4429, the user is presented with a Success screen and the process terminates. If instead, at step S4430, the user clicks on cancel request for activity, then at step S4431, the user is presented with a Success screen stating the request was canceled, and the process terminates.

[0351] The PIC system provides for Batch File updates. Figure 34 illustrates the PIC Batch service. Files are received over either a secure private network (SWIFTNet) 34.4 or using an existing private CONNECT:Direct 34.1 network to a

Bulk File Disk Storage **34.2**. The files are then routed to a Bulk File Server **34.3**. The PICBatchService recognizes this file and processes it. The executable program:

- [0352] • Starts a Windows Service in FileMonitorService.vb using the System.ServiceProcess. ServiceBase Class. This includes methods to control starting, pausing, continuing and stopping the service.
- [0353] • Within the method OnStart of the Service class, StartMonitor is called. This instantiates an object from the MonitorFiles class in MonitorFiles.vb. This object will then use the System.IO.FileSystemWatcher Class to monitor the file system for new files.
- [0354] • FileSystemWatcher.WaitForChange is then called to wait for new files being created in the input directory. When a file is created the FileSystemWatcher notifies the service by calling MonitorFiles.OnFileCreated, which queues a request to a thread pool. Multithreading is managed by using the System.Threading.Thread Class in.NET
- [0355] • The Thread will process the incoming file calling ProcessFile in class Importer. After obtaining exclusive use, the File will be moved to a working directory and then validated for a recognizable File Name format. Valid File Names will continue to be processed. Unrecognized File names will be reported to the Event Log and the thread process will then end.
- [0356] • File formats are then validated for the Incoming Batch File.
- [0357] • If format is correct then update requests are sent to the PIC database by calling the Business Facade Project, The request will pass through the Business Rules and Data Access Projects, performing validations and updates into the PIC database.
- [0358] • All results are received from the Business Facade as a Dataset.

- [0359] • An Output file will be generated in a temporary working directory based on the results. When the file is complete it will be moved to the appropriate Output Directory.
- [0360] • All phases of the batch process will use exception and notification trace logs to track both incoming and outgoing batch files. It will use the System Diagnostic namespace which exposes methods that allow logging of trace messages. This also includes interfaces to MOM 2000 and Event logs.
- [0361] The file formats consist of a Header record, followed by multiple Detail Records, which is followed by a File Control Record. Tables 6 and 7 depict the PIC Input Load File Format and the PIC Output Load File Format, respectively.

Table 6: PIC INPUT LOAD FILE FORMAT:

| INPUT FILE HEADER | | | | | | |
|--------------------|------|-----|---------|----------|---|--|
| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS | |
| HEADER INDICATOR | N | 1 | 1 | * | Indicates File Header. Always must be "1" | |
| SENDING BANK R/T | N | 9 | 2-10 | * | R/T of Bank sending File | |
| RECEIVING BANK R/T | N | 9 | 11-20 | * | IC&S Routing Transit Number | |
| FILE CREATION DATE | N | 8 | 21-28 | * | Date File was created. Format YYYYMMDD | |
| FILE MODIFIER | N | 2 | 29-30 | * | File Sequence Identifier | |
| FILER | AN | 770 | 31-800 | | | |

INPUT FILE HEADER

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|-----------------------|------|----|---------|----------|---|
| DETAIL INDICATOR | N | 1 | 1 | * | Indicates File Header. Always must be "6" |
| PIC DETAIL ACTION | AN | 6 | 2-7 | * | Files Originated from Bank: CREATE = add a new PIC MODIFY = change an existing PIC TRANSFR = transfer a PIC from another bank CLOSE = deactivate a PIC REACT = reactivate a PIC |
| PIC IDENTIFIER | AN | 17 | 8-24 | | Input will be zero filled for CREATE |
| UPDATE EFFECTIVE DATE | N | 8 | 25-32 | | Date update is effective Zeros assumes next business date Format is YYYYMMDD |
| PIC STATUS | N | 2 | 33-34 | * | Must be "00" on Input File |
| EPN CAPABLE | N | 1 | 35 | * | Valid Values: 0 = No 1 = Yes |
| CHIPS CAPABLE | N | 1 | 36 | * | Valid Values: 0 = No 1 = Yes |
| BANK RT NUMBER | N | 9 | 37-45 | * | FED's rt for this bank, or Rt # for customers bank |
| DDA ACCOUNT NUMBER | AN | 34 | 46-79 | * | DDA account number EPN has maximum of 17 char (Left Justified) |
| TAXPAYER ID | AN | 12 | 80-91 | * | |

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|-------------------|------|----|---------|----------|---|
| NAME | ANS | 96 | 92-187 | * | Corporate or individual's name |
| ADDRESS 1 | ANS | 64 | 188-251 | * | First Address line |
| ADDRESS 2 | ANS | 64 | 252-315 | * | Second line of address Required if Address 3 is present |
| ADDRESS 3 | ANS | 64 | 316-379 | * | Third line of address Required if Address 4 is present |
| ADDRESS 4 | ANS | 64 | 380-443 | * | Fourth line of address |
| CITY | ANS | 32 | 444-475 | | |
| STATE/PROV | ANS | 32 | 476-507 | | |
| POSTAL CODE | AN | 11 | 508-518 | | |
| COUNTRY | ANS | 3 | 519-521 | * | Values defined by ISO-3166 3-letter codes. |
| SIC BUSINESS CODE | N | 4 | 522-525 | | Standard Industrial Classification codes |
| DUNES ID | AN | 10 | 526-535 | | D&B D-U-N-S Numbers |
| BIC CODE | N | 11 | 536-546 | | SWIFT BIC Code |
| BEI CODE | N | 11 | 547-557 | | SWIFT Business Entity Identifier |
| IBAN CODE | N | 34 | 558-591 | | International Bank Account Number |
| WIRE PAY N/A 1 | ANS | 35 | 592-626 | | CHIPS/Fed Wire Payment Name/Address 1 |

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|----------------|------|----|---------|----------|---------------------------------------|
| WIRE PAY N/A 2 | ANS | 35 | 627-661 | | CHIPS/Fed Wire Payment Name/Address 2 |
| WIRE PAY N/A 3 | ANS | 35 | 662-696 | | CHIPS/Fed Wire Payment Name/Address 3 |
| WIRE PAY N/A 4 | ANS | 35 | 697-731 | | CHIPS/Fed Wire Payment Name/Address 4 |
| FILLER | AN | 69 | 732-800 | | Filled with spaces |

ADDITIONAL NOTES:

Required Fields lists above for the Input Detail Records apply only for "Create", "Modify" and "Transfer" Actions. The fields required for "Close", and "Reactivate" are as follows:

CLOSE:

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|-------------------|------|-----|---------|----------|--|
| DETAIL INDICATOR | N | 1 | 1 | * | Indicates File Header. Always must be "6" |
| PIC DETAIL ACTION | AN | 6 | 2-7 | * | CLOSE = deactivate a PIC |
| PIC IDENTIFIER | AN | 17 | 25-32 | * | Date update is effective Zeros assumes next business date Format is YYYYMMDD |
| FILLER | AN | 768 | 33-800 | | |

REACTIVATE:

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|------------------|------|----|---------|----------|---|
| DETAIL INDICATOR | N | 1 | 1 | * | Indicates File Header. Always must be "6" |

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|-----------------------|------|----|---------|----------|--|
| PIC DETAIL ACTION | AN | 6 | 2-7 | * | REACT - Reactivate a PIC |
| PIC IDENTIFIER | AN | 17 | 8-24 | * | Input will be zero filled for CREATE |
| UPDATE EFFECTIVE DATE | N | 8 | 25-32 | * | Date update is effective Zeros assumes next business date Format is YYYYMMDD |

FILLER

AN

768

33-80

INPUT FILE CONTROL

| FIELD NAME | TYPE | SZ | COLUMNS | REQUIRED | COMMENTS |
|----------------------|------|-----|---------|----------|--|
| CONTROL INDICATOR | N | 1 | 1 | * | Indicates File Control. Always must be "9" This is Last record on File. |
| TOTAL DETAIL RECORDS | AN | 793 | 8-800 | | |

Table 7: PIC OUTPUT LOAD FILE FORMAT:

| OUTPUT FILE HEADER | | | | |
|--------------------|------|-----|---------|---|
| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
| HEADER INDICATOR | N | 1 | 1 | Indicates File Header. Always must be "1" |
| SENDING BANK R/T | N | 9 | 2-10 | IC&S Routing Transit Number |
| RECEIVING BANK R/T | N | 9 | 11-20 | R/T of Bank sending File |
| FILE CREATION DATE | N | 8 | 21-28 | Date File was created. Format YYYYMMDD |
| FILE MODIFIER | N | 2 | 29-30 | File Sequence Identifier |
| REJECTED FILE IND | N | 1 | 31 | Values: 0 - File was Accepted 1 - File was Rejected |
| REJECTED TEXT | AN | 50 | 32-71 | This will list the Error message if an input file is rejected |
| FILER | AN | 729 | 72-800 | |

OUTPUT FILE DETAIL

| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
|-----------------------|------|----|---------|--|
| DETAIL INDICATOR | N | 1 | 1 | Indicates File Header. Always must be "6" |
| PIC DETAIL ACTION | AN | 6 | 2-7 | Valid values are ACCEPT - PIC detail action request accepted by IC&S ERROR - PIC detail action request denied, Associated text begins in column 751 |
| PIC IDENTIFIER | AN | 17 | 8-24 | Zero filled for CREATE that was denied |
| UPDATE EFFECTIVE DATE | N | 8 | 25-32 | Date update is effective Format YYYYMMDD |
| PIC STATUS | N | 2 | 33-34 | Valid Values: "00" - Create request was denied "01" - Open (Active) "02" - Closed (Inactive) "03" - Transfer in Process "04" - Transfer being contested "05" - Awaiting approval via Web "06" - Request scheduled for Effective date to be Implemented "99" - Suspended |
| EPN CAPABLE | N | 1 | 35 | Valid Values: 0 = No 1 = Yes |

| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
|--------------------|------|----|---------|---|
| CHIPS CAPABLE | N | 1 | 36 | Valid Values: 0 = No 1 = Yes |
| BANK RT NUMBER | N | 9 | 37-45 | FED's rt for this bank, or Rt # for customers bank |
| DDA ACCOUNT NUMBER | AN | 34 | 46-79 | DDA account number EPN has maximum of 17 char (Left Justified) |
| TAXPAYER ID | AN | 12 | 80-91 | |
| NAME | ANS | 96 | 92-187 | Corporate or individual's name |
| ADDRESS 1 | ANS | 64 | 188-251 | First Address line |
| ADDRESS 2 | ANS | 64 | 252-315 | Second line of address Required if Address 3 is present |
| ADDRESS 3 | ANS | 64 | 316-379 | Third line of address Required if Address 4 is present |
| ADDRESS 4 | ANS | 64 | 380-443 | Fourth line of address |
| CITY | ANS | 32 | 444-475 | |
| STATE/PROV | ANS | 32 | 476-507 | |
| POSTAL CODE | AN | 11 | 508-518 | |
| COUNTRY | ANS | 3 | 519-521 | Values defined by ISO-3166 3-letter codes. |
| SIC BUSINESS CODE | N | 4 | 522-525 | Standard Industrial Classification codes |

| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
|----------------|------|----|---------|--|
| DUNS ID | AN | 10 | 526-535 | D&B D-U-N-S Numbers |
| BIC CODE | N | 11 | 536-546 | SWIFT BIC Code |
| BEI CODE | N | 11 | 547-557 | SWIFT Business Entity Identifier |
| IBAN CODE | N | 34 | 558-591 | International Bank Account Number |
| WIRE PAY N/A 1 | ANS | 35 | 592-628 | CHIPS/Fed Wire Payment Name/Address 1 |
| WIRE PAY N/A 2 | ANS | 35 | 627-661 | CHIPS/Fed Wire Payment Name/Address 2 |
| WIRE PAY N/A 3 | ANS | 35 | 662-696 | CHIPS/Fed Wire Payment Name/Address 3 |
| WIRE PAY N/A 4 | ANS | 35 | 697-731 | CHIPS/Fed Wire Payment Name/Address 4 |
| FILLER | AN | 19 | 732-750 | Filled with spaces |
| ERROR TEXT | AN | 50 | 751-800 | Text for error if detail record was rejected |

OUTPUT FILE CONTROL

| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
|----------------------|------|----|---------|---|
| CONTROL INDICATOR | N | 1 | 1 | Indicates File Control. Always must be "9" This is Last record on File. |
| TOTAL DETAIL RECORDS | N | 6 | 2-7 | Total Number of Detail Records on File. |
| ACCEPTED RECORDS | N | 6 | 8-13 | Total number of PIC updates Accepted |
| REJECTED RECORDS | N | 6 | 14-19 | Total number of PIC updates Rejected |

| FIELD NAME | TYPE | SZ | COLUMNS | COMMENTS |
|---------------|------|-----|---------|---|
| TOTAL RECORDS | N | 6 | 20-25 | Total Number of Records on File. Count of All records including Header, Details, and Control. |
| FILLER | AN | 775 | 26-800 | |

[0362] A Configuration File named PICBatchService.EXE.Config is also be included. This is an XML file that must be included in the same path as the executable program. Included in <App Settings> are elements for:

- Connection String
- Input File Path
- Incoming Working Path
- Temporary Output File Path
- Output File Path for Connect Direct
- Output File Path for SWIFT

[0363] File Titles for processing:

Input Batch Files:

Path - \'Input Path\'RT\Source\File Name

Input Path - From Configuration File

Source - Connect Direct = CD

SWIFT = SWIFT

File Name - UINYYYMMDD99.DAT

Where: YYYMMDD - File Creation Date

99 - File Sequence (Start 1 thru 99)

Output Batch Files:

Path - \'Output Path\'RT\Source\File Name

Output Path - From Configuration File

Source - Connect Direct = CD

SWIFT = SWIFT

File Name - UOUTYYMMDD99.DAT

Where: YYYMMDD - File Creation Date

99 - File Sequence (Start 1 thru 99)

[0364] Each file for an RT should have a unique Creation Date and Modifier.

[0365] There will be fields designated as required and optional Requests to "Close" and "Reactivate" will have a limited number of required fields.

[0366] Detail records requesting a "Transfer" will only be accepted if the entire file contains Transfer requests.

[0367] Output files will always return the entire PIC record.

[0368] These will be Binary files in ASCII format without CR & LF characters.

[0369] The IC&S RT will be used as the Receiving RT for incoming batch files and the Sending RT for Outgoing batch files.

[0370] A file that is not formatted properly will be rejected.

[0371] Possible file format errors include:

1. Missing File Header Record
2. Invalid Sending RT for File
3. Invalid Receiving RT for File
4. File Control is not last record
5. Total detail records on File Control is incorrect

[0372] Rejected files will be returned as an Output File with

1. Reject File Indicator of "I" and Error Text in Output File Header.
2. Accepted Records of 0 in Output File Control
3. Rejected Records count will be equal to Total Detail Records in File

Control.

[0373] Detail errors will be designated by "ERROR" in the PIC Detail Action field of the Output batch file and added to the Rejected totals count.

[0374] Specific error text will be included as part of the Output File Format where noted.

[0375] While the present invention has been described with respect to what is presently considered to be the preferred embodiment, it is to be understood that the invention is not limited to the disclosed embodiment. To the contrary, the invention is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims. The scope of the

following claims is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structures and functions.

WHAT IS CLAIMED:

1. A secure electronic payment method for effecting electronic payment between an originator's account and a beneficiary's account, safeguarding banking and account information, while utilizing existing payment systems, said method comprising the steps of:
 - generating a system routing number and a payment identification code (PIC) relating to the beneficiary's account information;
 - distributing a list of payment identification codes to the existing payment system and financial institutions owning the account related to the payment identification codes on the list;
 - the originator receiving a system routing number and the beneficiary's PIC number;
 - the originator communicating a payment instruction to a financial institution of the originator, wherein the payment instruction includes the system routing number beneficiary's payment identification code;
 - the originator's financial institution receiving the payment instruction from the originator, wherein if the received PIC matches the originator's financial institution internal list of PICs, the originator's financial institution performs an "on us" transaction;
 - transmitting payment instruction to an existing payment system in a case when the received PIC does not match originator's financial institution internal list of PICs;
 - the existing payment system validating the received PIC against a PIC database, wherein if the PIC is invalid, the payment instruction is returned to originator's financial institution;
 - converting the PIC and system routing number to a receiving payment instruction in a case when the PIC is a valid PIC, wherein the receiving

payment instruction includes the beneficiary's financial institution's routing number and the beneficiary's account number;

the existing payment system transmitting the receiving payment instruction to a financial institution of the beneficiary;

the beneficiary's financial institution crediting the beneficiary's account if no problem exists, and otherwise returning a receiving payment instruction to the existing payment system; and

upon receipt of the returned receiving payment instruction by the existing payment system, the existing payment system translating the receiving payment instruction into the payment instruction prior to transmitting the payment instruction back to originator's financial institution,

wherein the payment identification code is unique to each beneficiary's account.

2. A secure electronic payment method according to claim 1, wherein the existing payment system is an Electronic Payment Network (EPN).

3. A secure electronic payment method according to claim 1, wherein the existing payment system is a Clearing House Interbank Payments System (CHIPS).

4. A secure electronic payment method according to claim 1, further comprising the steps of:

generating a database, wherein the database includes the name and routing number information of the beneficiary's financial institution, the payment identification code (PIC) which uniquely identifies the beneficiary's account information; and

distributing a copy of the database to the existing payment systems.

5. A secure electronic payment method according to claim 4, wherein the database is a secure database managed by a trusted third party.

6. A web-based payment method for effecting electronic payment between an originator's account and a beneficiary's account utilizing existing payment systems, said method comprising the steps of:

generating a payment identification code and system routing number uniquely identifying account information of the beneficiary;

distributing the payment identification code and the beneficiary's account information relating to the payment identification code to the existing payment systems;

the beneficiary transmitting payment identification code to the originator;

in response to a payment order from beneficiary, the originator transmitting payment instruction to the financial institution of the originator, wherein payment instruction includes the payment identification code of the beneficiary, and amount to be paid;

the originator's financial institution processing and transmitting payment instruction to existing payment system to effect electronic funds transfer of funds;

the existing payment system converting payment identification code included in payment instruction to beneficiary's account information and forwarding converted payment instruction to beneficiary's financial institution; and

the beneficiary's financial institution effecting electronic funds transfer on the basis of converted payment instruction by crediting beneficiary's account,

wherein communication between the originator, the financial institution of the originator, existing payment system, beneficiary's financial institution, and beneficiary is accomplished at least partly via the Internet.

7. A web-based payment method according to claim 6, further comprising the step of the existing payment system distributing a list payment identification codes to financial institutions owning the account related to the payment identification code.

8. A web-based payment method according to claim 7, further comprising the steps of:

upon receipt of the payment instruction, the originator's financial institution checking the list of payment identification codes to determine whether the beneficiary's payment identification code is on the list; and

in a case where the payment identification code of the beneficiary is on the list performing an "on us" transaction.

9. A web-based payment method claim 8, wherein the existing payment system is an Electronic Payment Network (EPN).

10. A web-based payment method claim 8, wherein the existing payment system is a Clearing House Interbank Payments System (CHIPS).

11. A web-based payment method claim 8, further comprising the steps of:

generating a database, wherein the database includes the name and routing number information of the beneficiary's financial institution, the payment identification code (PIC) which uniquely identifies the beneficiary's account information; and

distributing a copy of the database to the existing payment systems.

12. A web-based payment method claim 11, wherein the database is a secure database managed by a trusted third party.

13. A secure electronic payment method between a consumer and a biller facilitated through an existing payment system, comprising the steps of:

- generating a payment identification code unique to biller's account information;
- distributing the payment identification code to biller and the existing payment system;
- the biller communicating the payment identification code to the consumer;
- the consumer electronically transmitting a payment instruction via consumer's financial institution to the existing payment system, said payment instruction comprising information indicating at least source of consumer's account information, payment amount, and biller's payment identification code;
- the existing payment system validating the payment identification code of the biller;
- upon validating the payment identification code of the biller, the existing payment system converting the payment identification code of the biller included in the payment instruction into biller's account information which includes routing number of biller's financial institution;
- transmitting converted payment instruction to biller's financial institution;
- applying a credit to biller's account in an amount corresponding to the payment amount included in the payment instruction; and
- applying a debit to consumer's account in an amount corresponding to the payment amount included in the payment instruction.

14. A secure electronic payment method according to claim 13, wherein the existing payment system is an Electronic Payment Network (EPN).

15. A secure electronic payment method according to claim 13, wherein the existing payment system is a Clearing House Interbank Payments System (CHIPS).

Fig. 1 Integration of Supply and Financial Chains

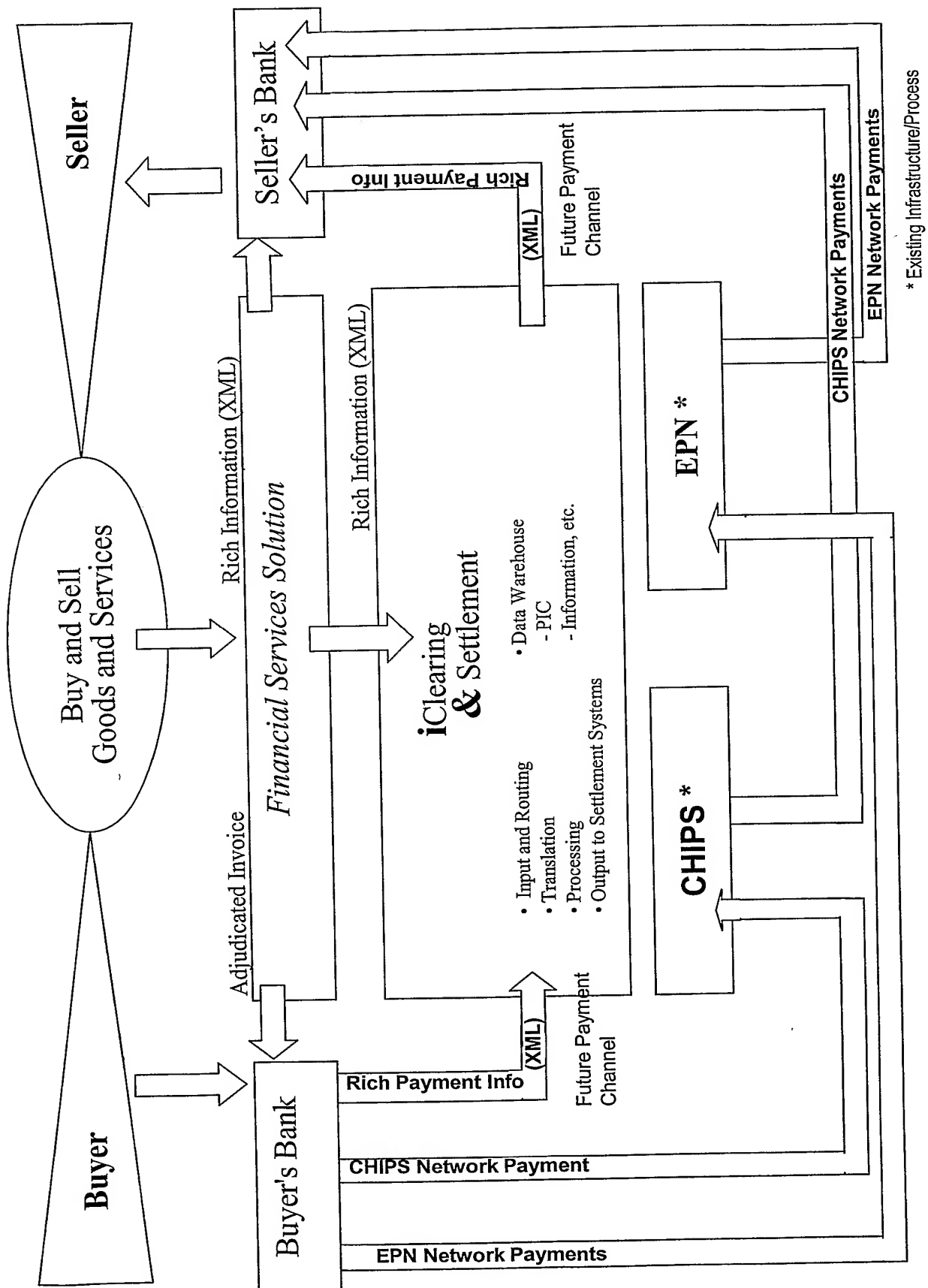


Figure 2

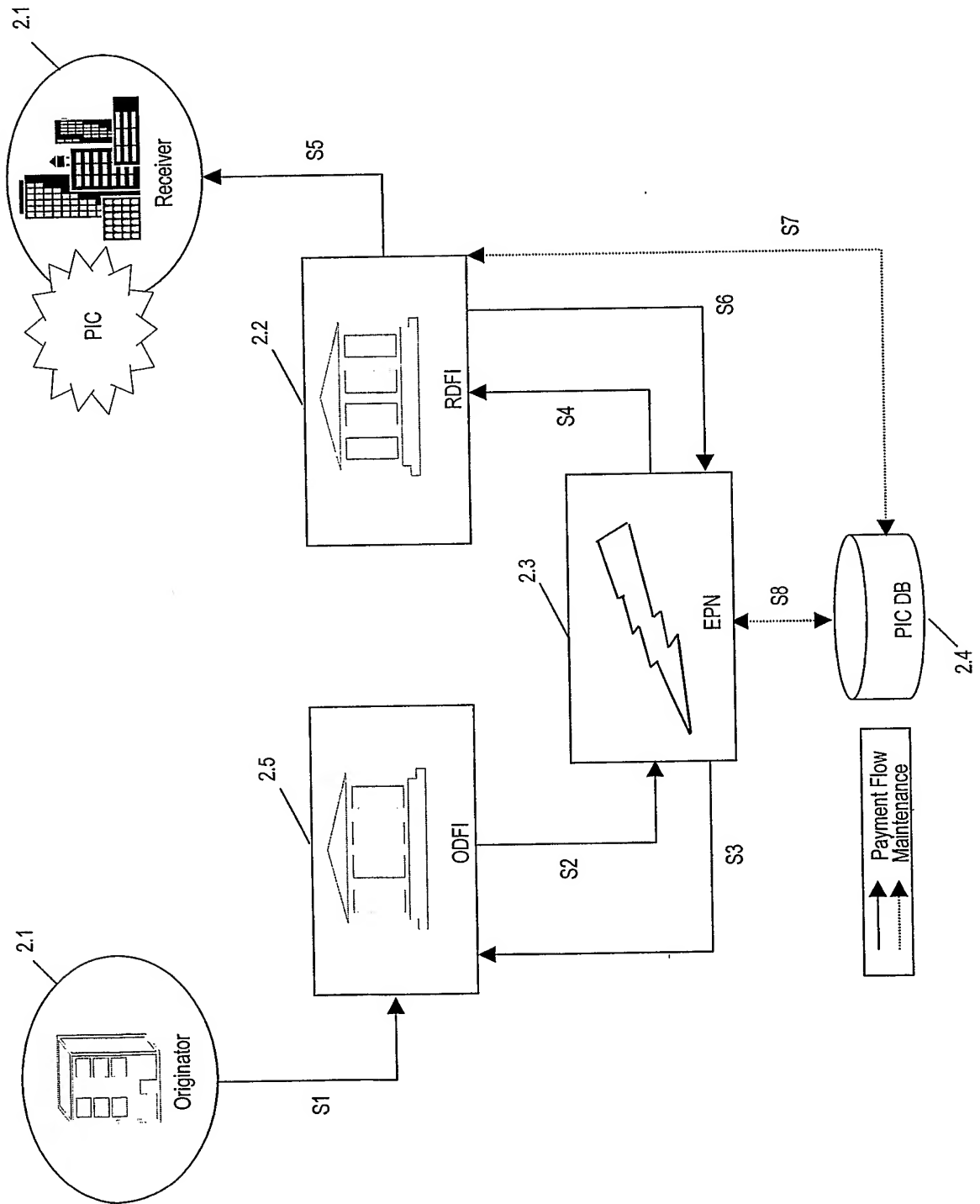


Figure 3

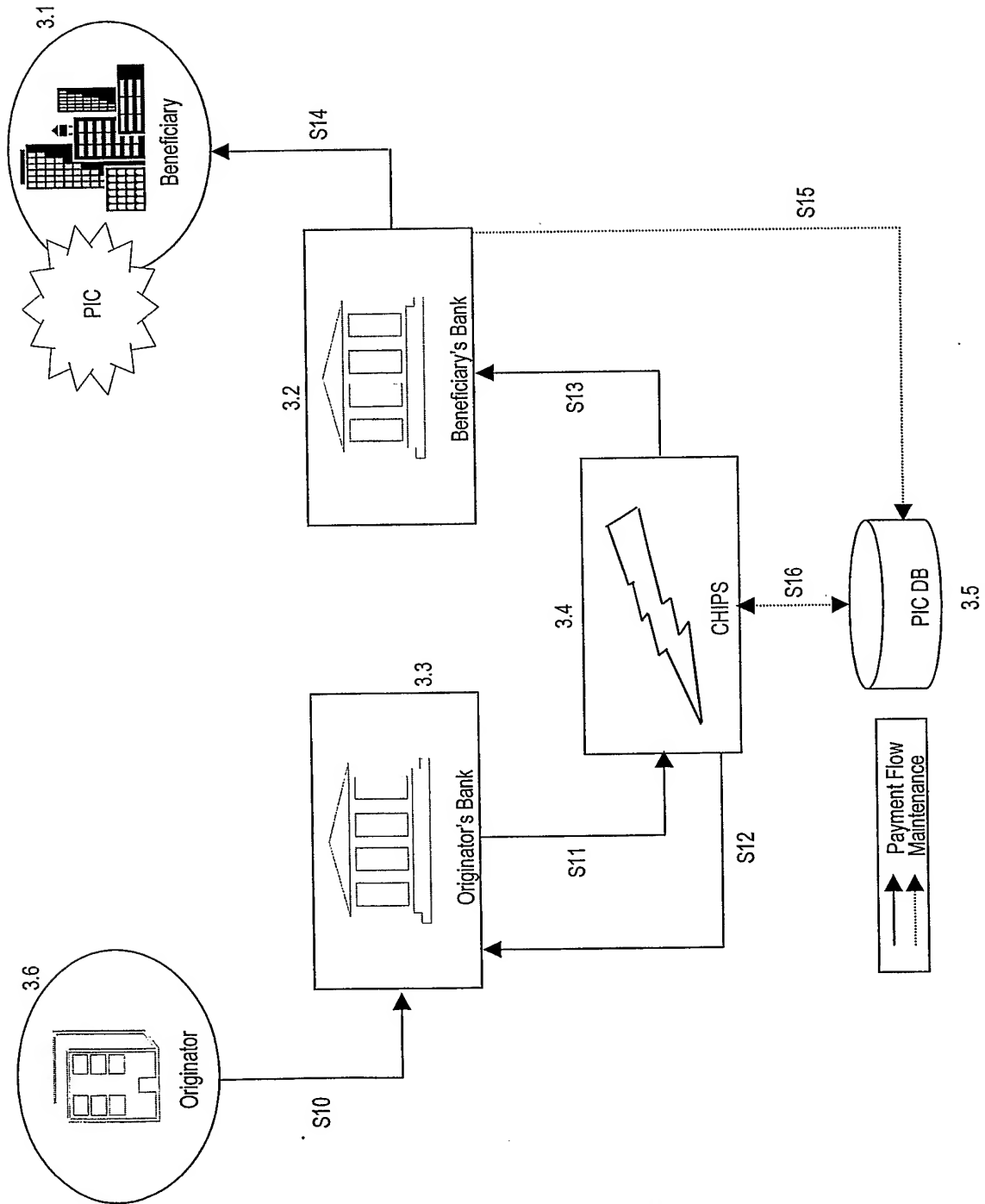


Figure 4

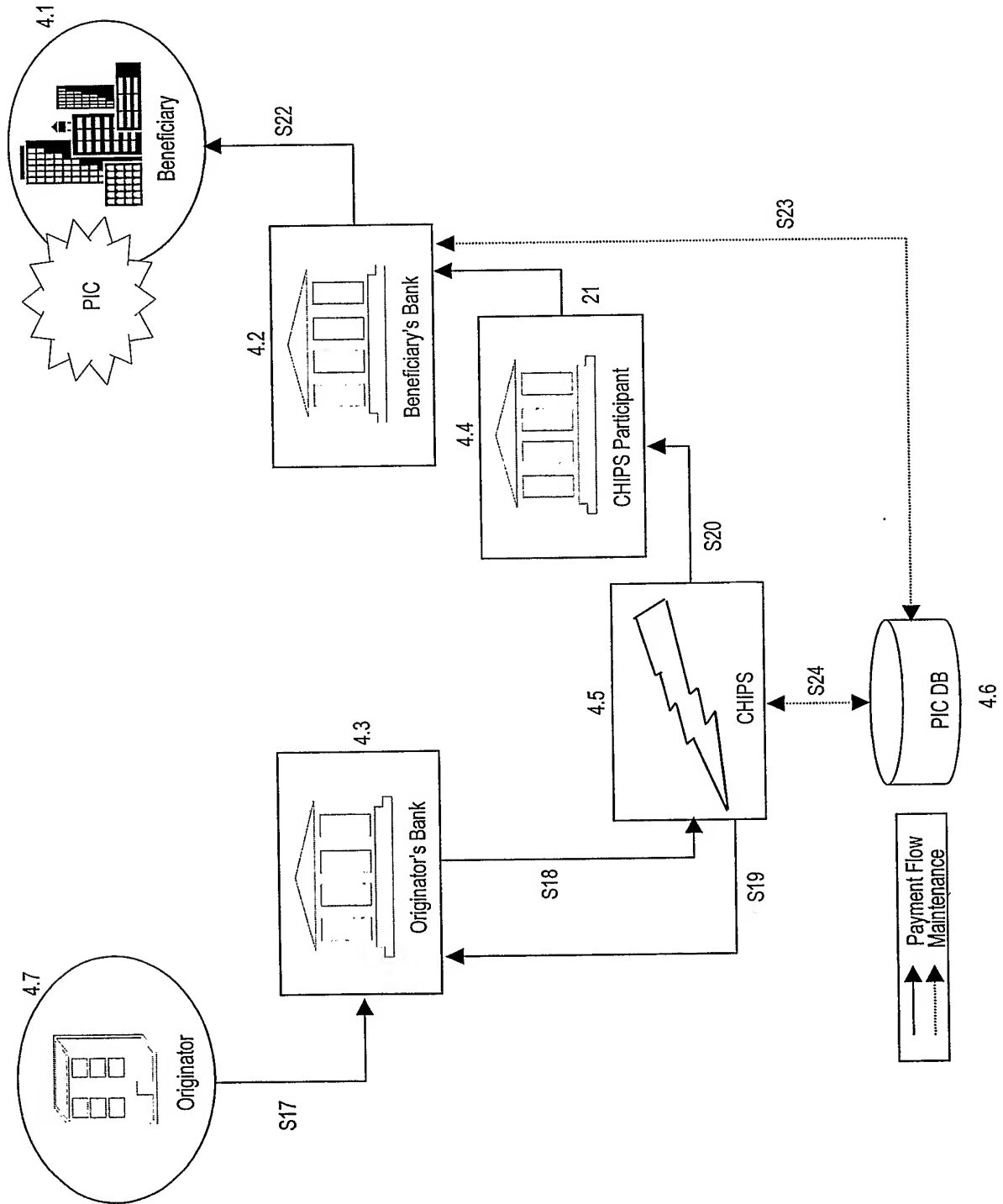


Fig. 5 PIC Sample Cash Management Entry

http://www.ph-support.com/demos/payments2001demo/common/mainWebWCMview.htm - Microsoft Internet Explorer provide...

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites Full Screen Mail Size Print Messenger

Address http://www.ph-support.com/demos/payments2001demo/common/mainWebWCMview.htm

Links Home News Services Headlines Agreement Instant Internet Instant Search My Presario

Database List Batch List ACH Validator

Message Board

ACH Payments

Account Transfers

Cash Concentration

Wire Activity

Reporting

Stop Payments

Setup

Maintenance

Vendor Name: Quatex Inc.

Vendor ID: 11111

Amount: \$35,000.00

Payment Description: Invoice # 785454585 10/25/2000

Vendor Bank ID: 11111119

Vendor Bank Acct No. 4549871

Last Payment Date

Hold Transfer

5.1

5.2

5.3

5.4

5.5

5.6

IC&S Routing Number

Universal Payment ID Code

Create Prenote:

Prenote Date: 02/09/2000

Exit

Updated: 09/21/2000 Version 1

OHierarchical Menu Trees Created

Start || || Sen... FW... IC&... Micr... htt... Pre...

Internet

SE

11:32 AM

Figure 6

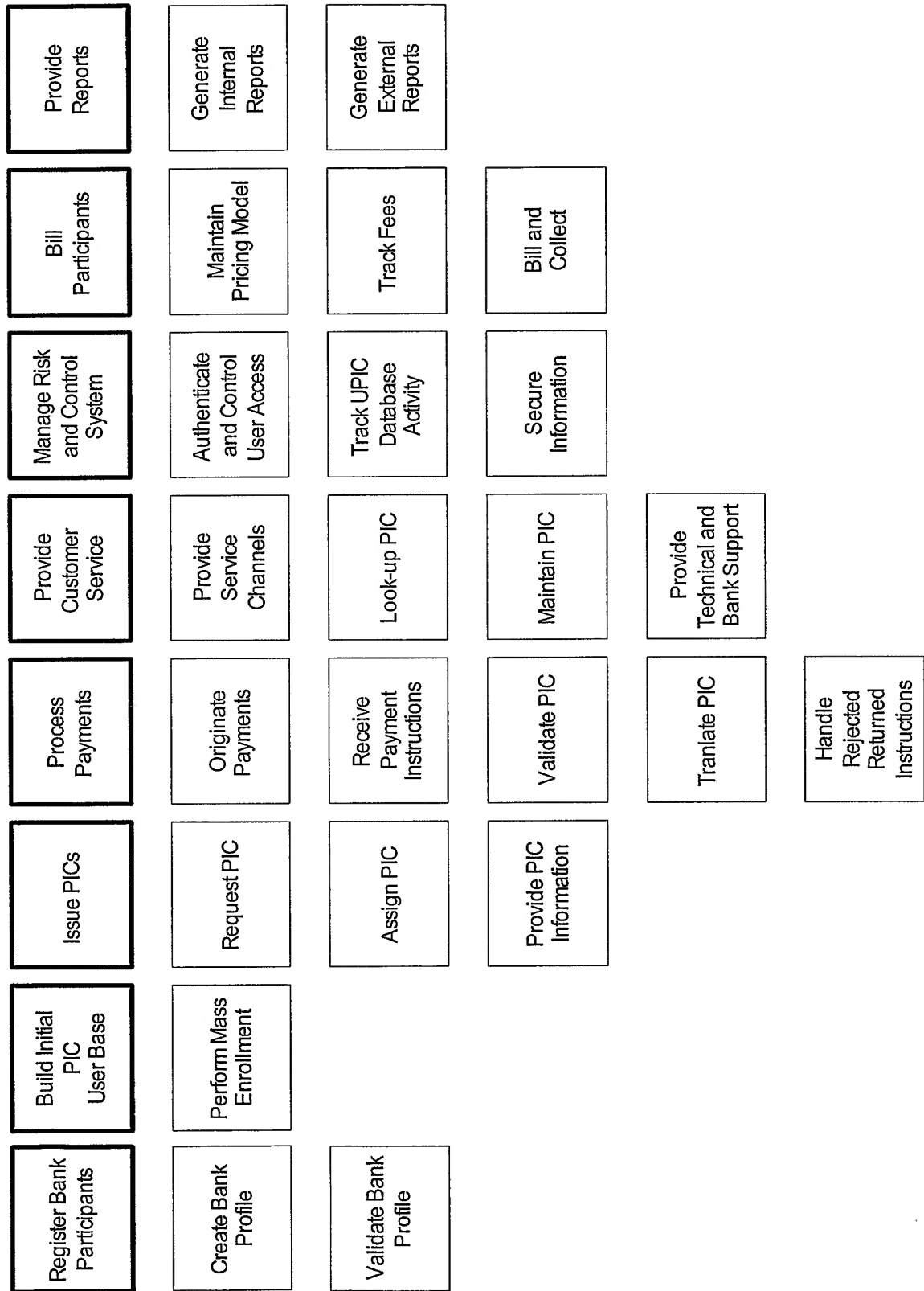


Figure 7 Entity Relation Diagram

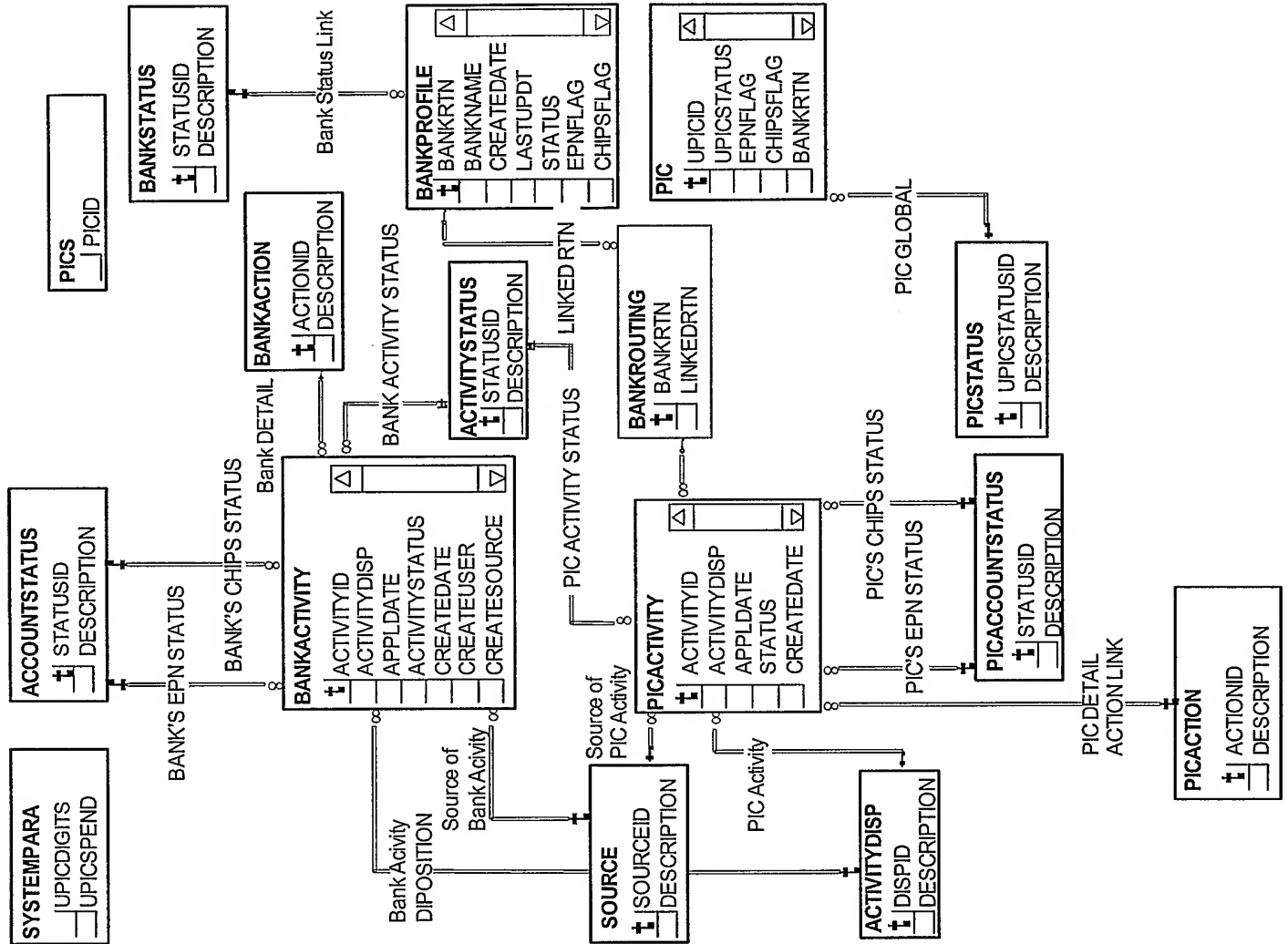
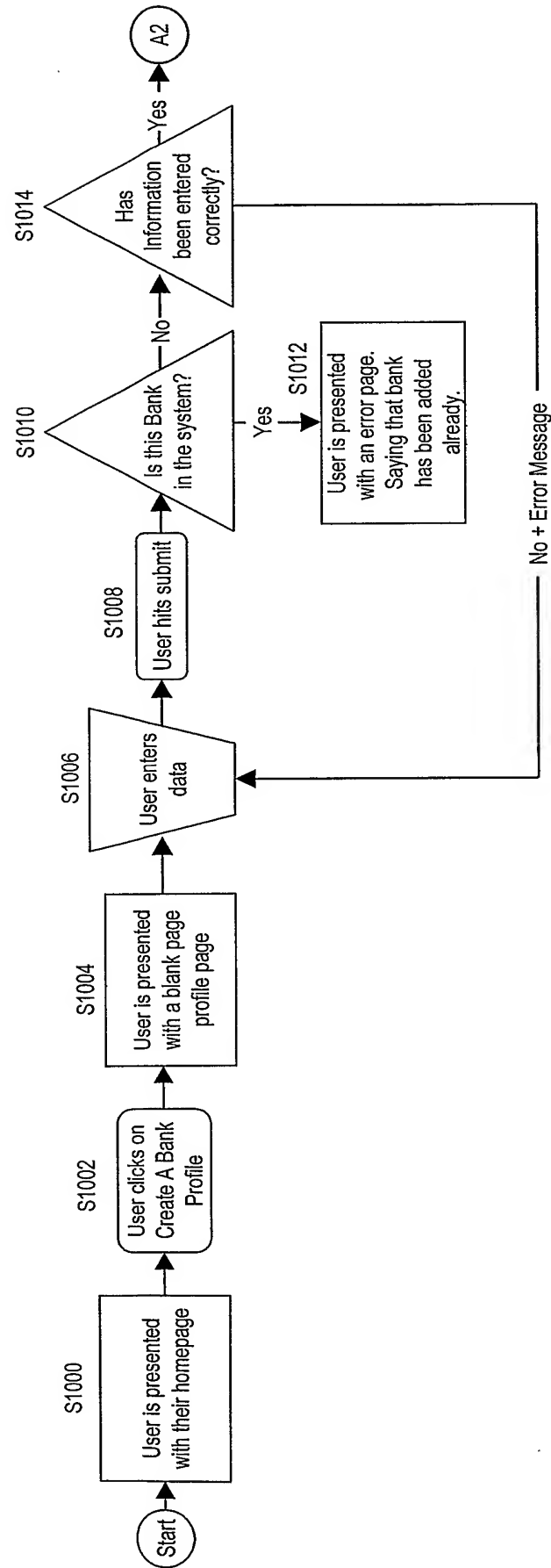


Fig. 8 Create A Bank Profile



Legend



= Page

= Link



= System process



= More in section A

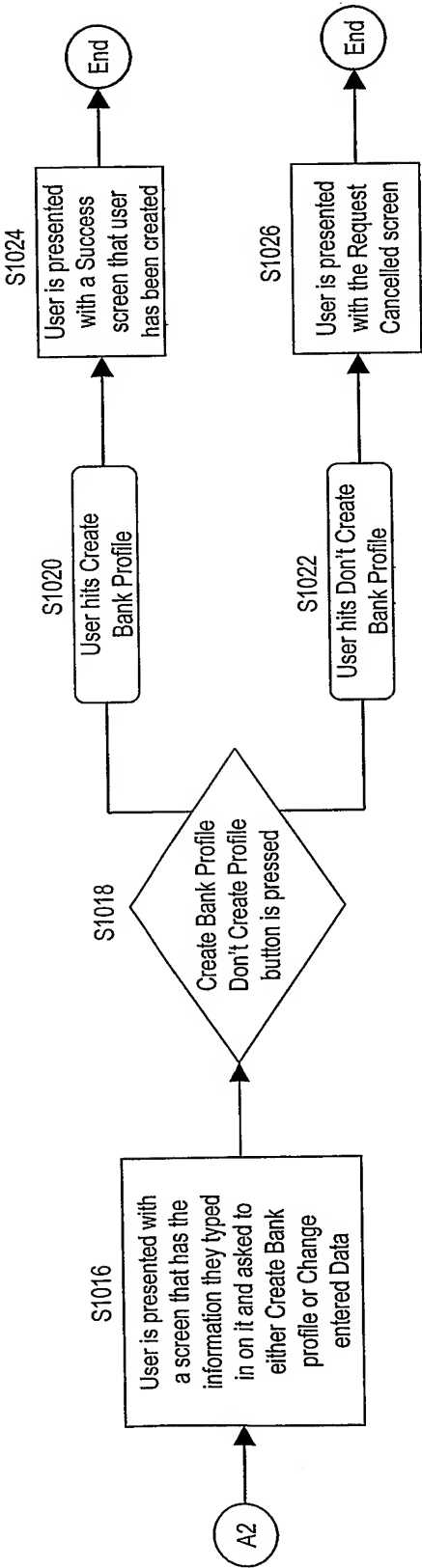


= Manual process



= User Decision

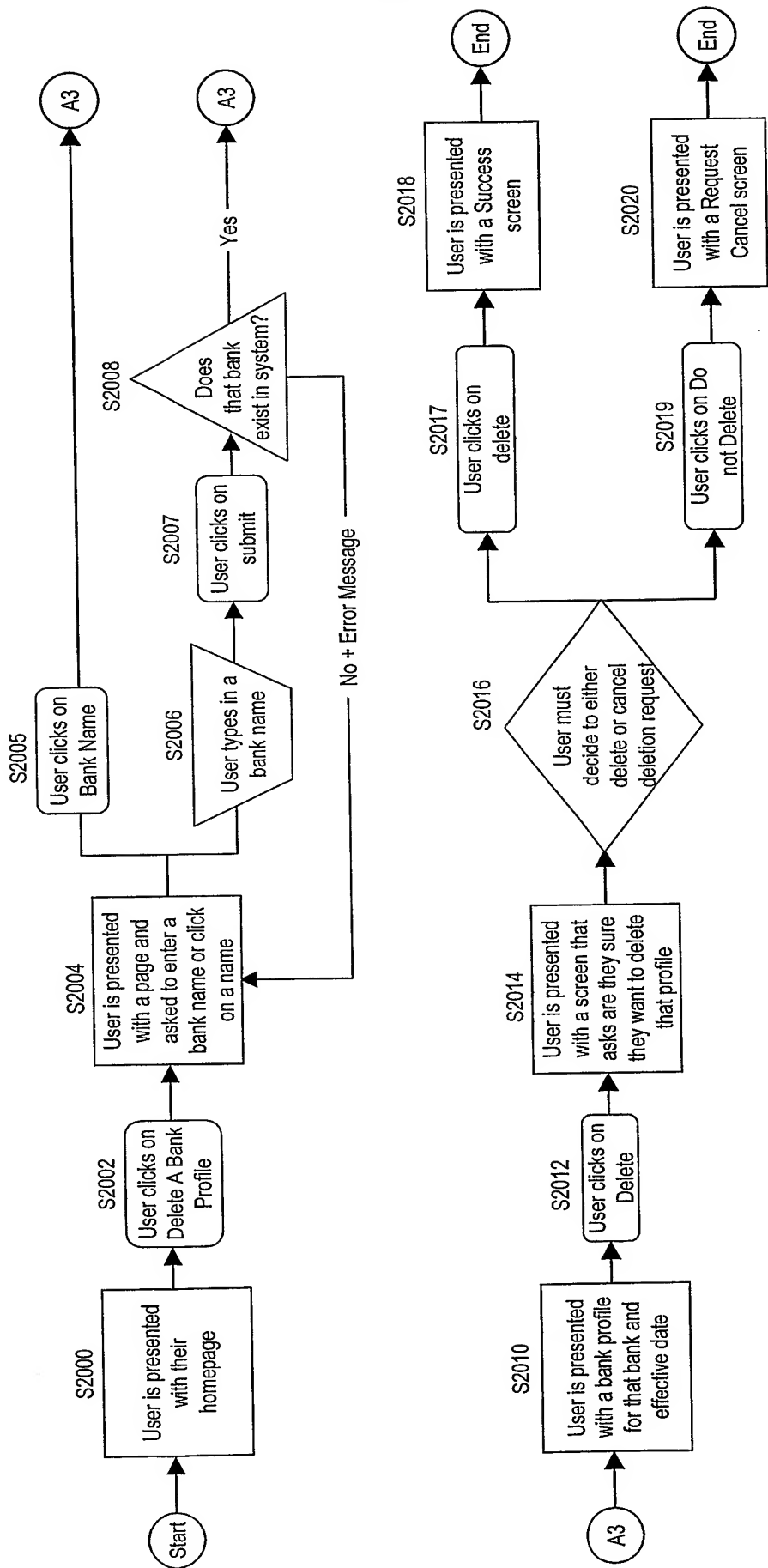
Fig. 9 Create A Bank Profile (cont'd)



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 10 Delete A Bank Profile



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 11 Modify A Bank Profile

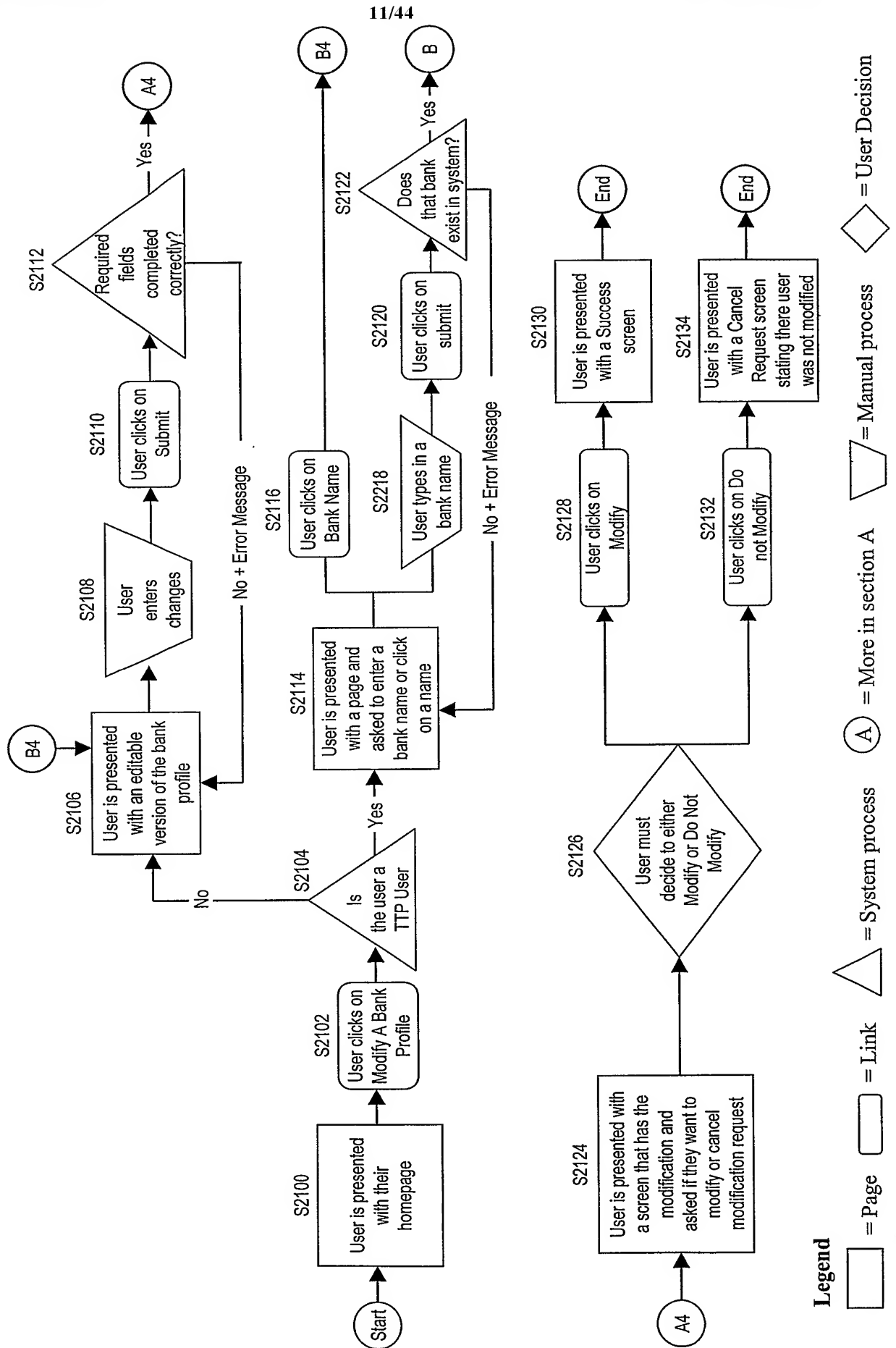


Fig. 12 Change a Route

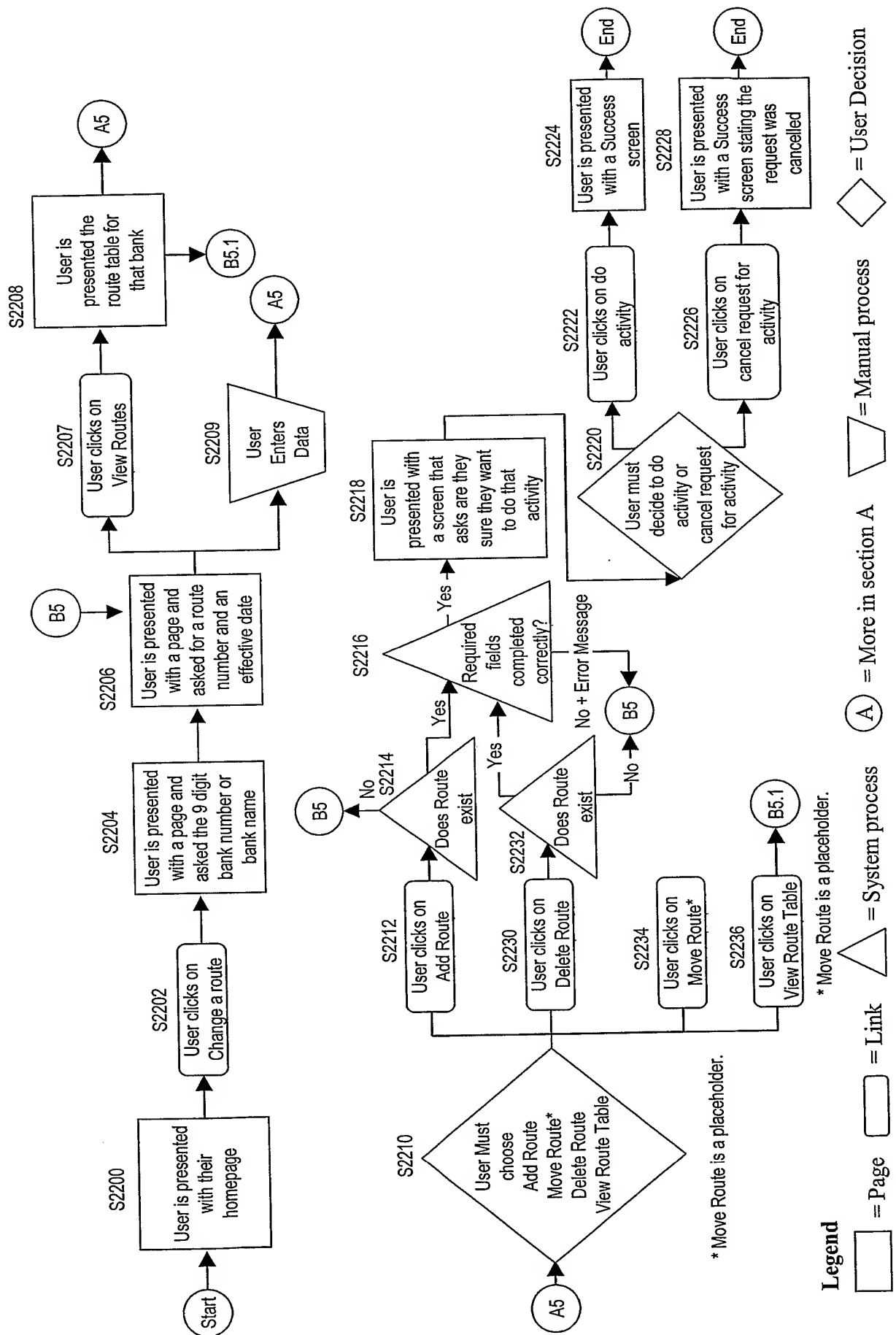


Fig. 13 View PIC Activity Log

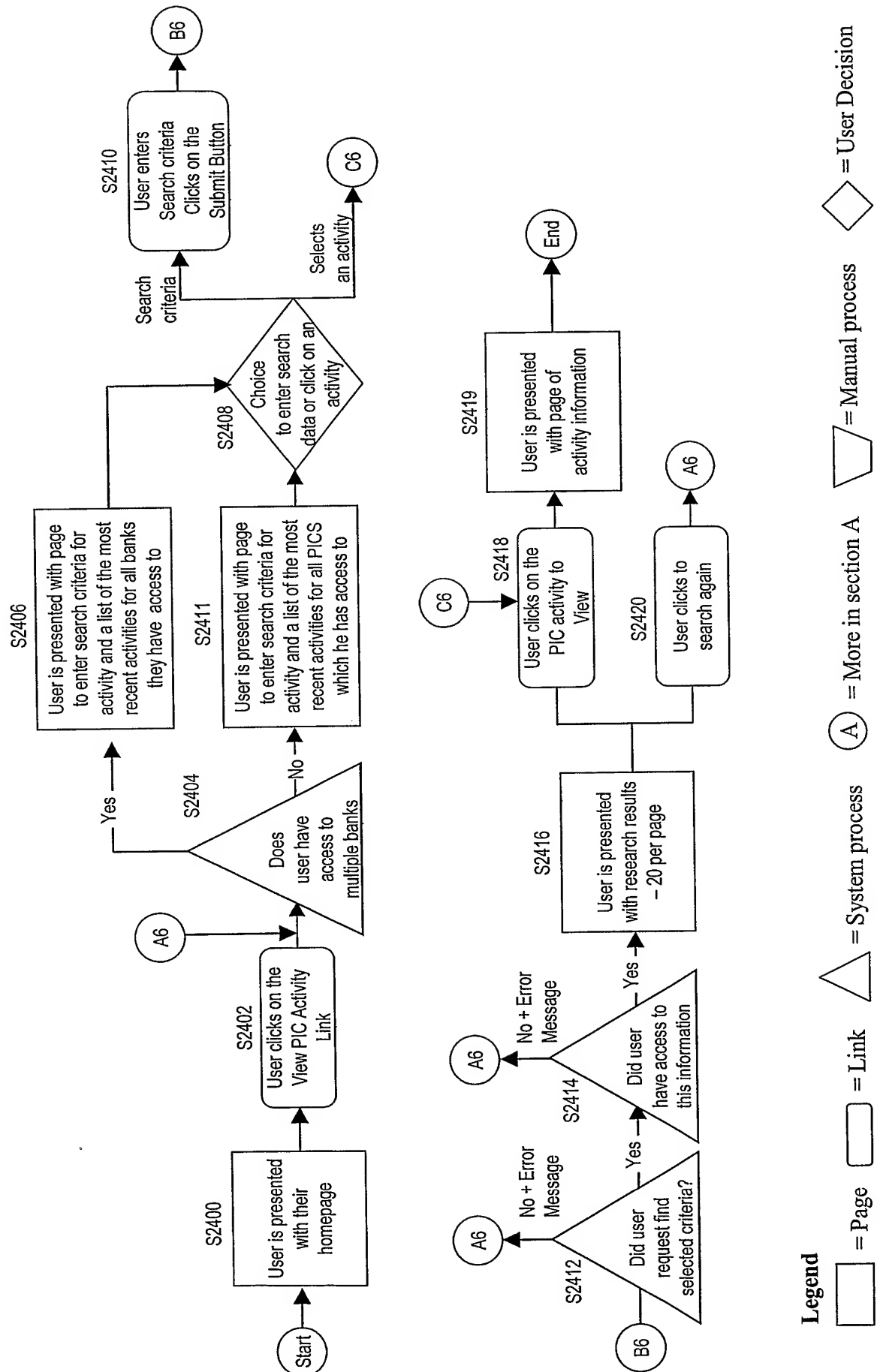
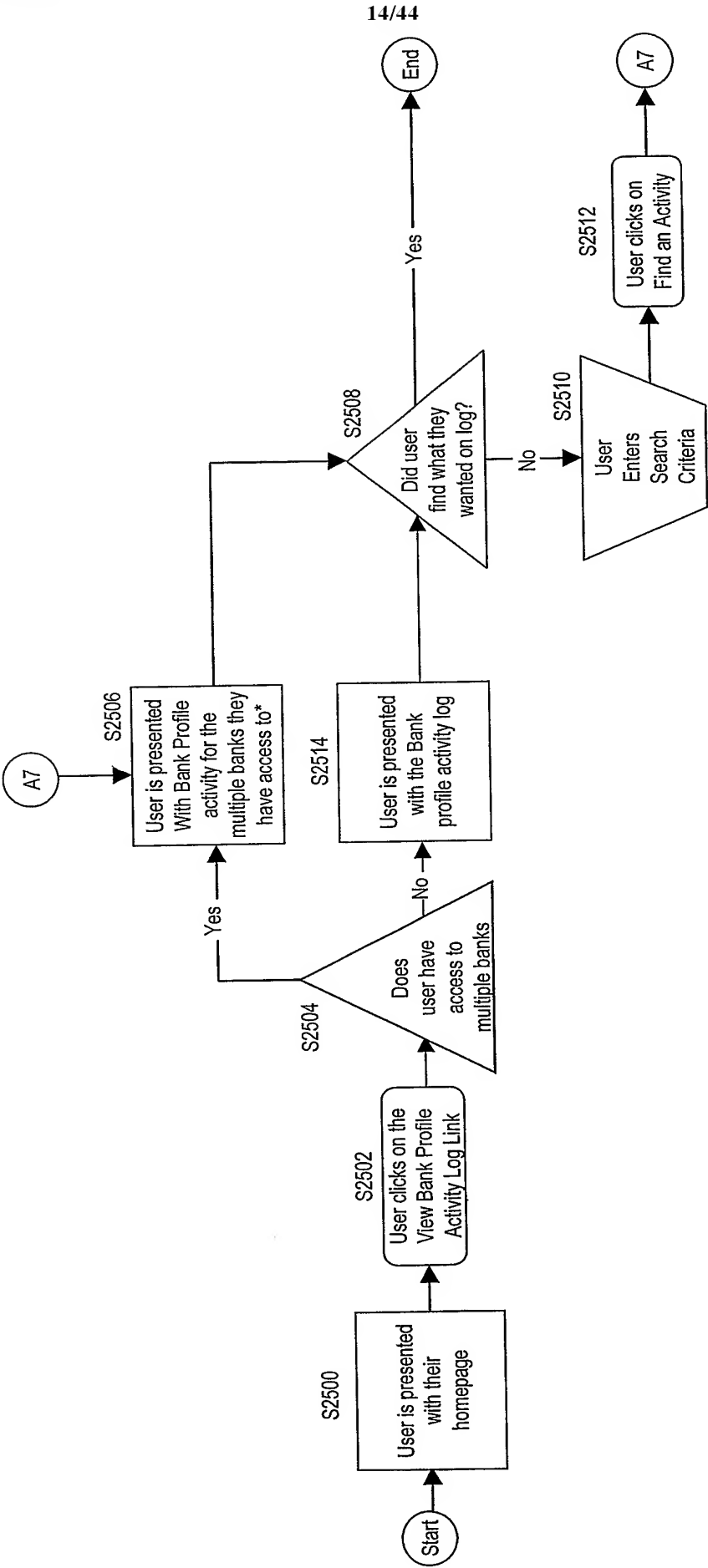


Fig. 14 View Bank Profile Activity Log

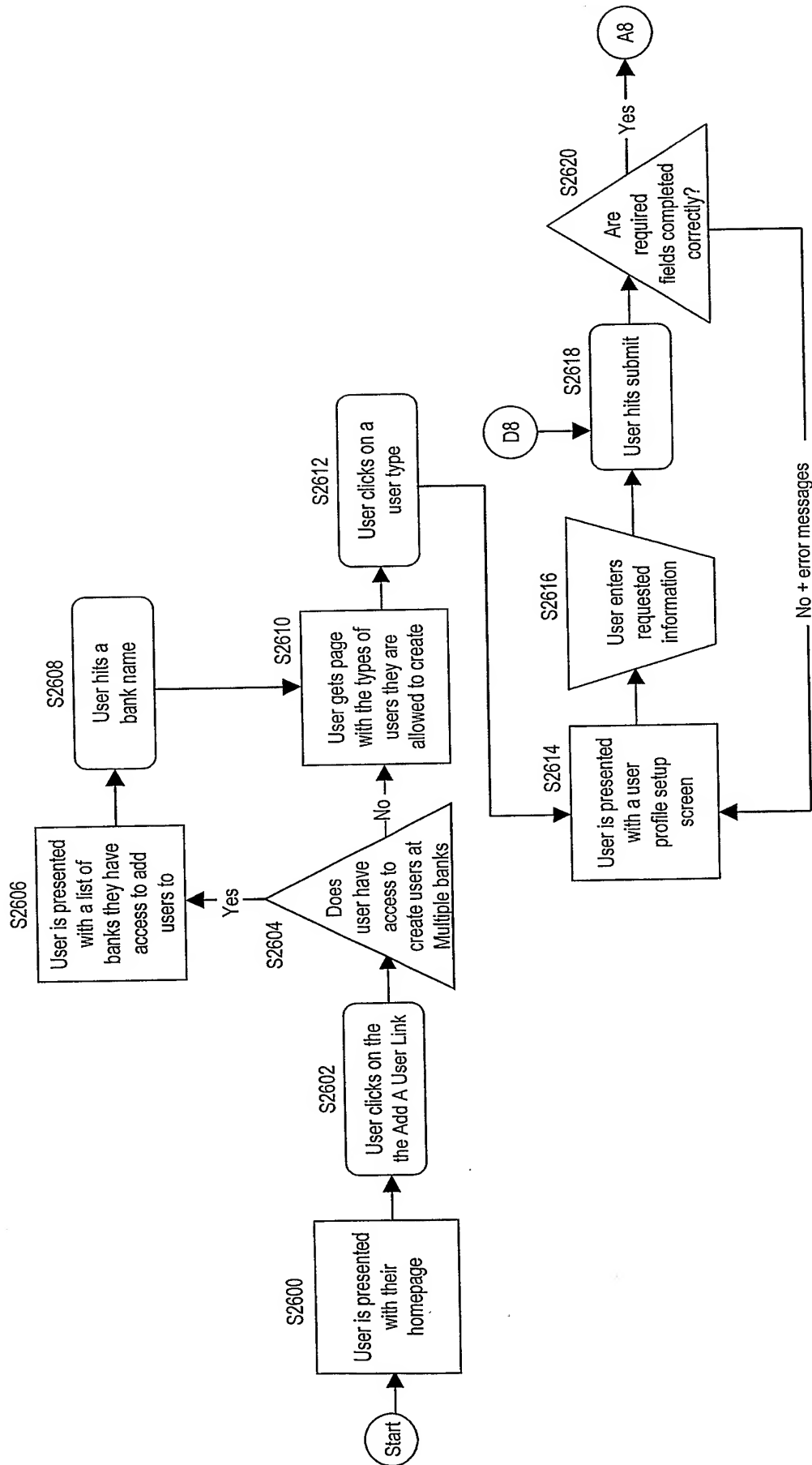


*Note: If the activity is in the future and the user has right to perform that activity, a CANCEL button should appear to allow for the cancellation of that pending activity.

Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 15A Create A User



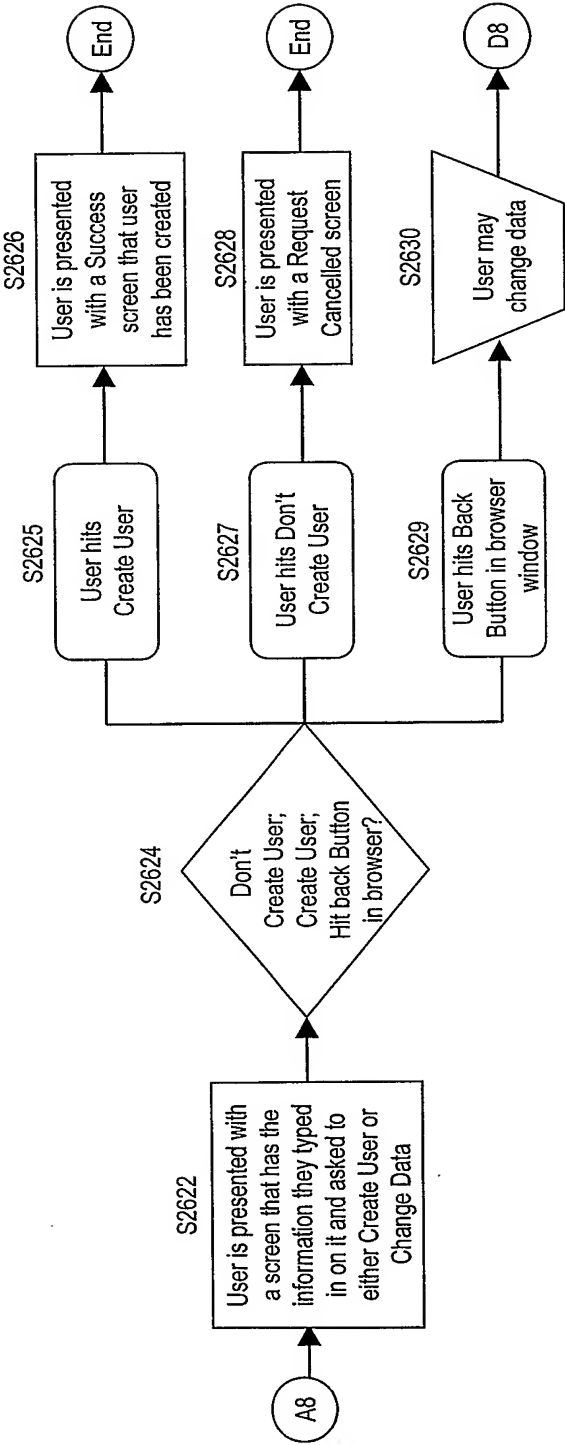
Legend



[] = Page
 () = Link
 △ = System process
 ▤ = Manual process
 ◇ = User Decision

(A) = More in section A
 (A8) = More in section A

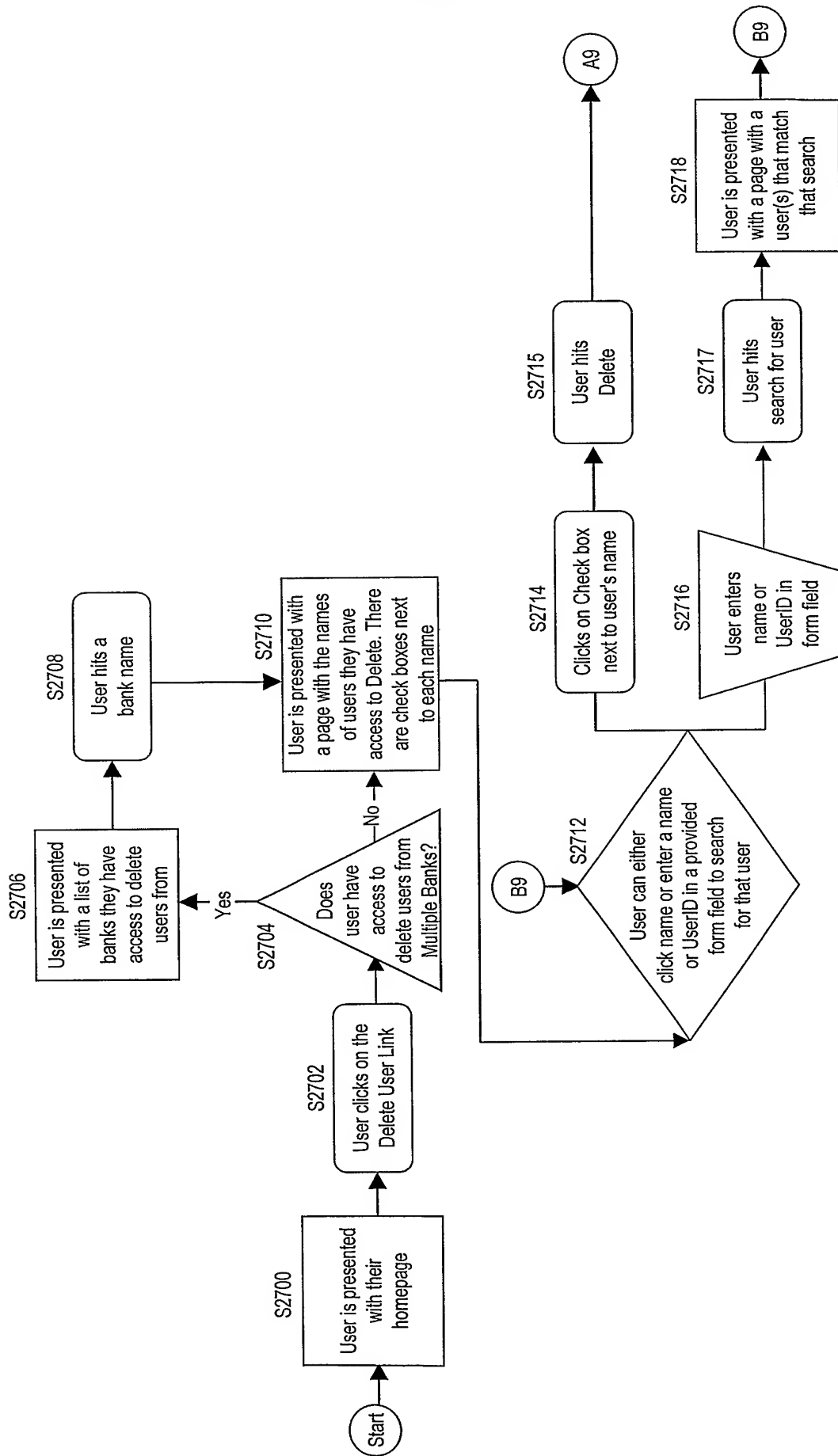
Fig. 15B Create A User



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 16A Delete A User

**Legend**

= Page
 = Link
 = System process
 = More in section A
 = Manual process
 = User Decision

Fig. 16B Delete A User

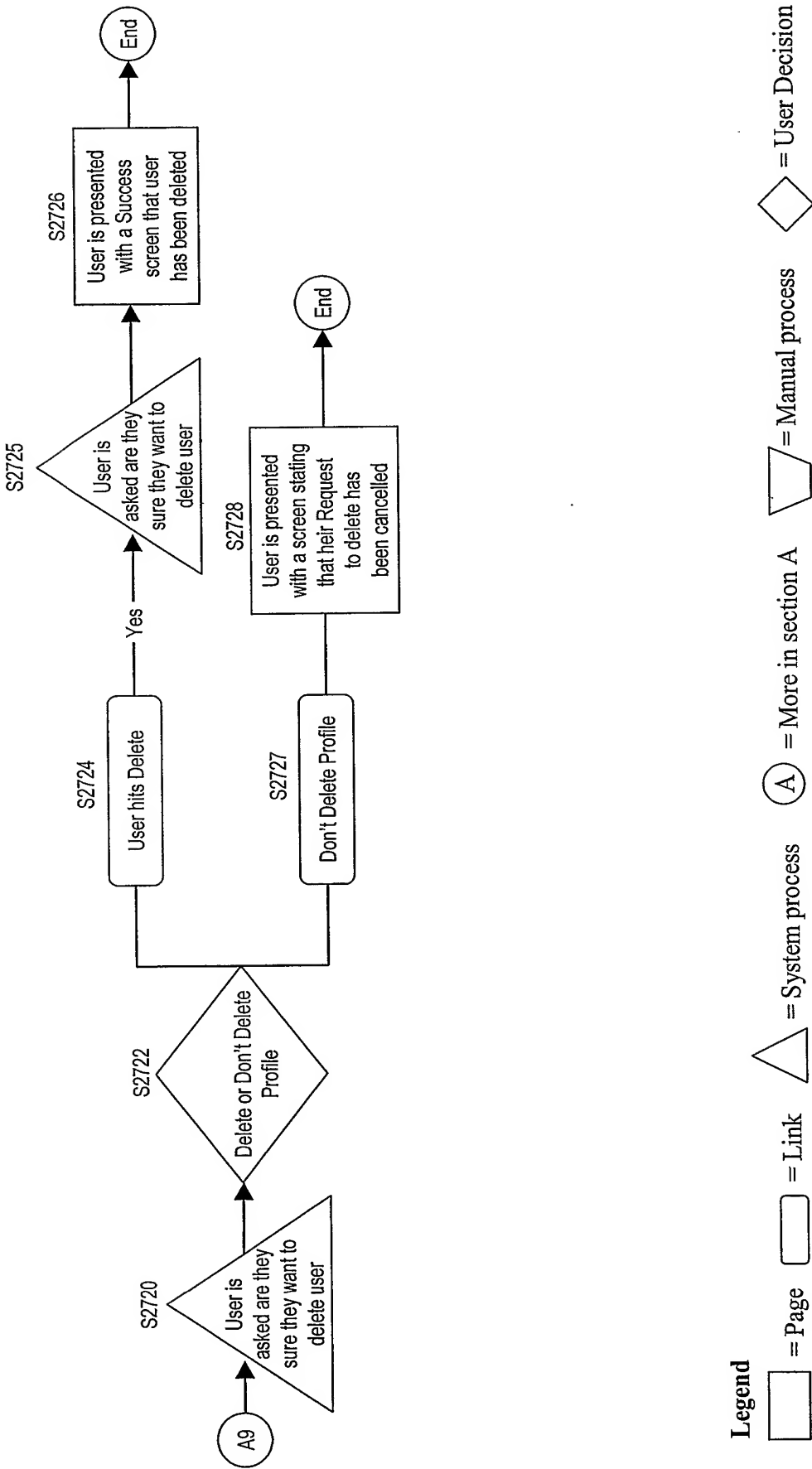


Fig. 17A Modify A User

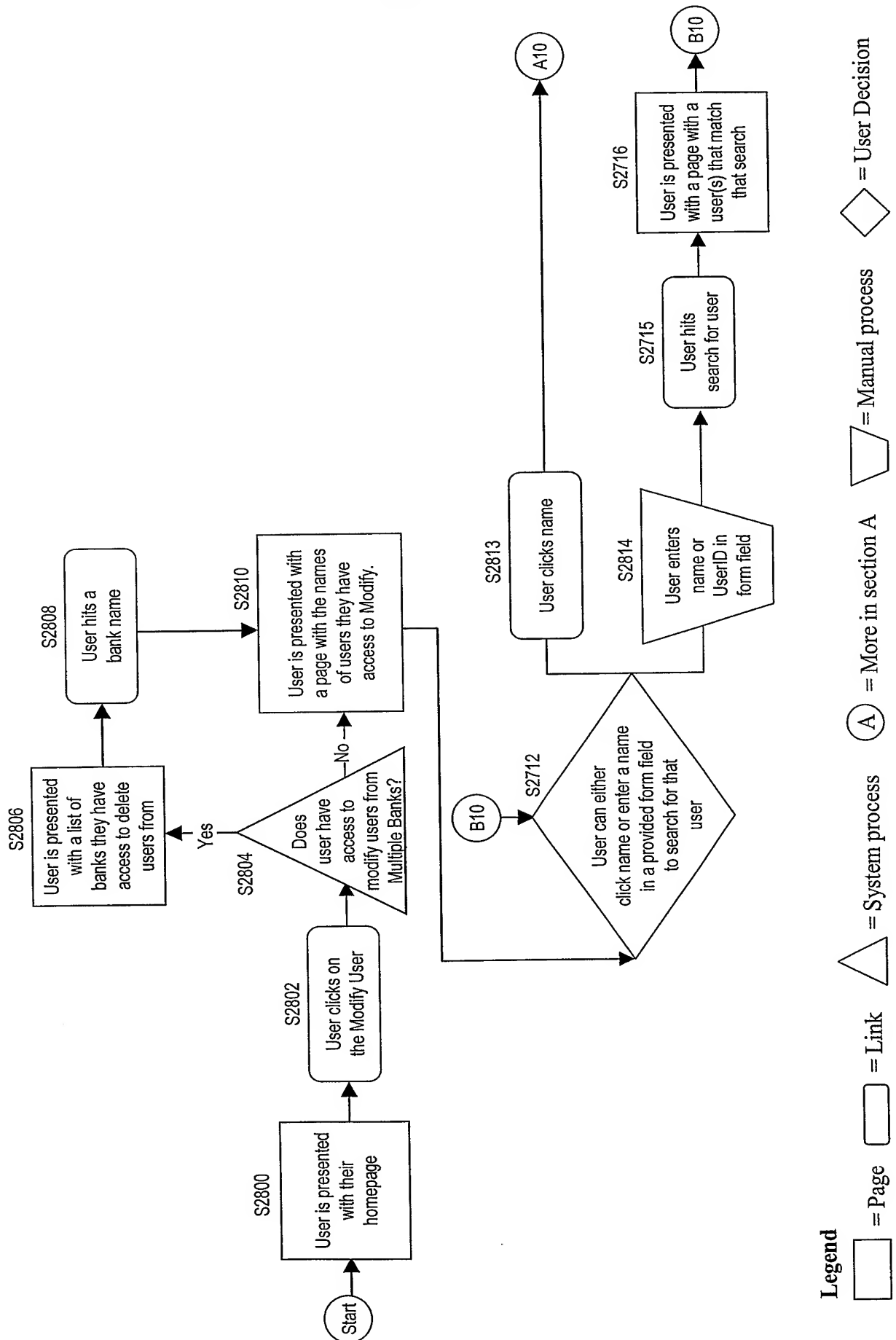
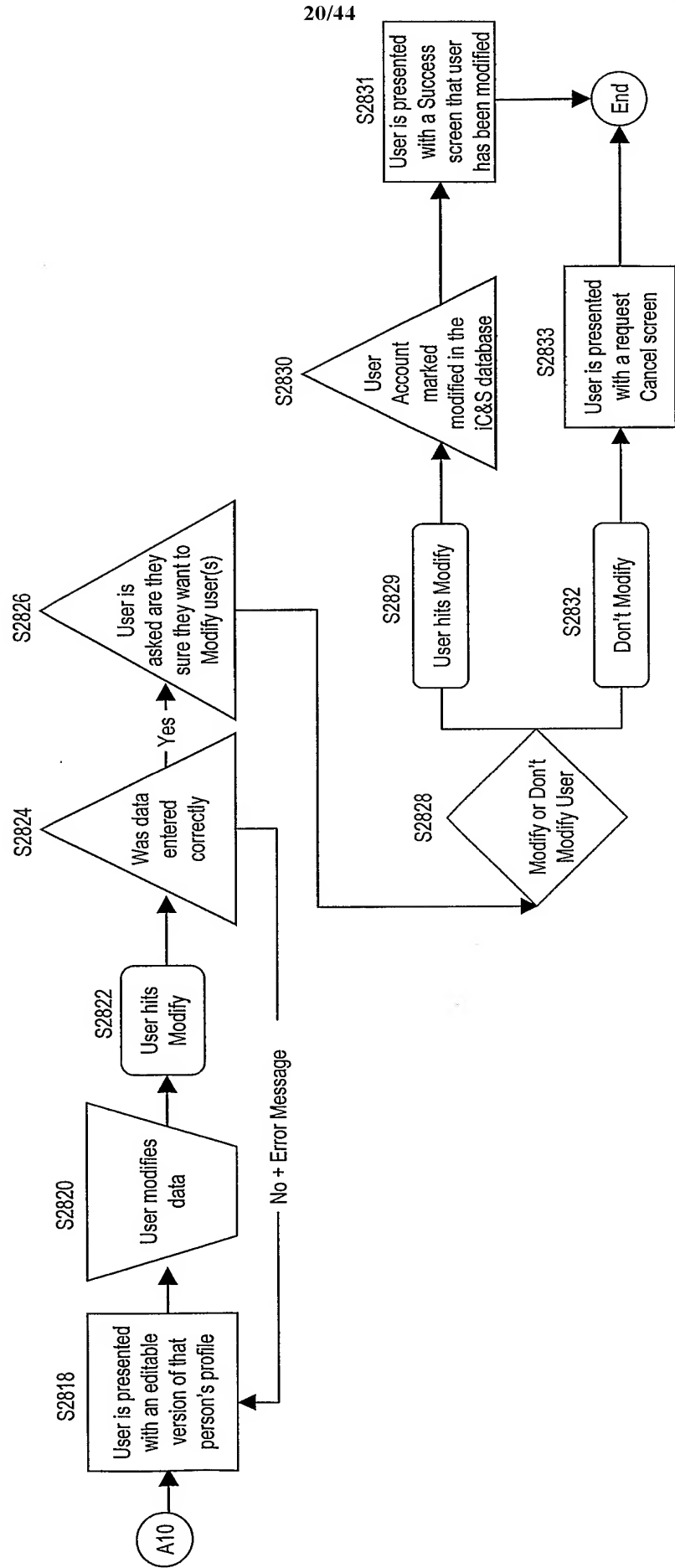


Fig. 17B Modify A User



Legend

Page

Link

System process

More in section A

Manual process

User Decision

Fig. 18 View A User Profile

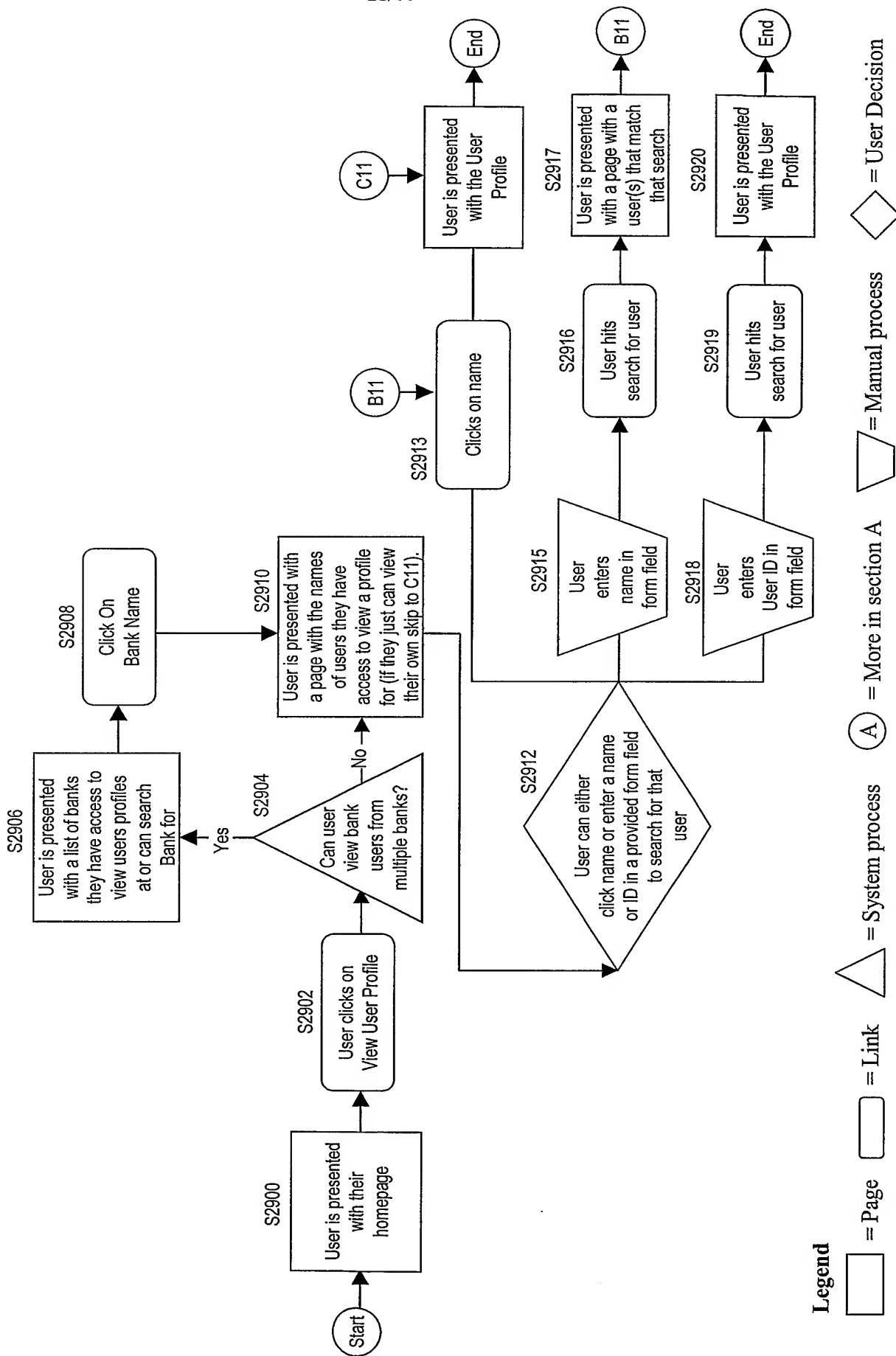
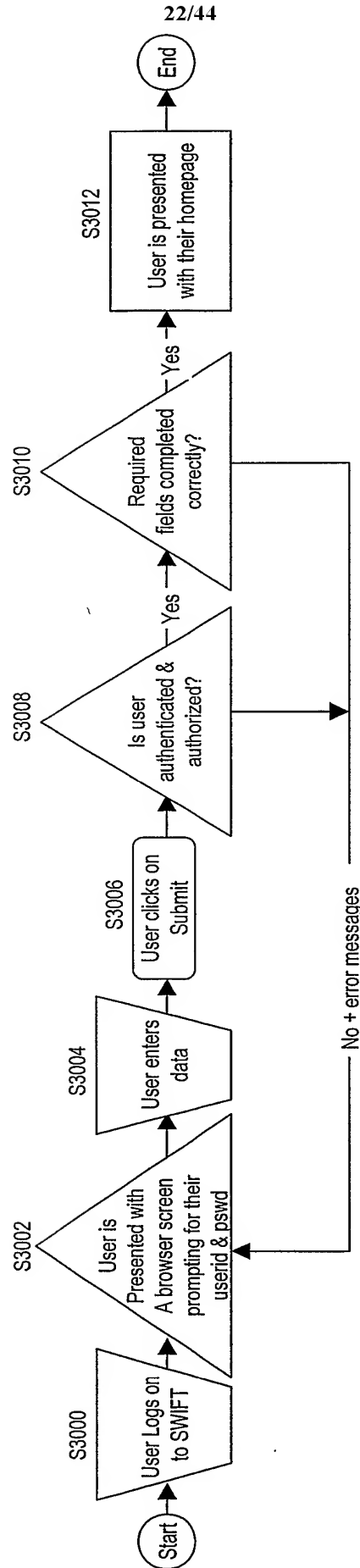


Fig. 19 Log On To The PIC System



Legend

| | | | | | | | | | |
|--|--------|--|------------------|--|---------------------|--|------------------|--|-----------------|
| | = Page | | = System process | | = More in section A | | = Manual process | | = User Decision |
|--|--------|--|------------------|--|---------------------|--|------------------|--|-----------------|

Fig. 20 Change a Password

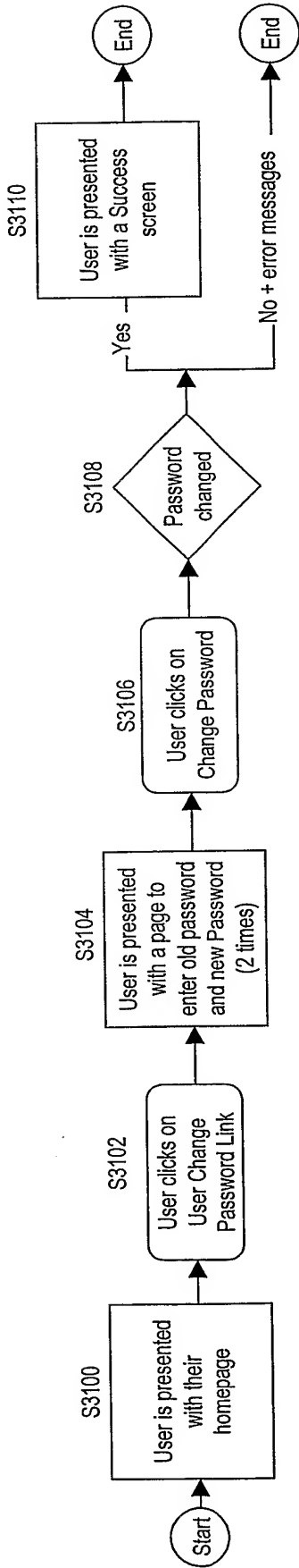
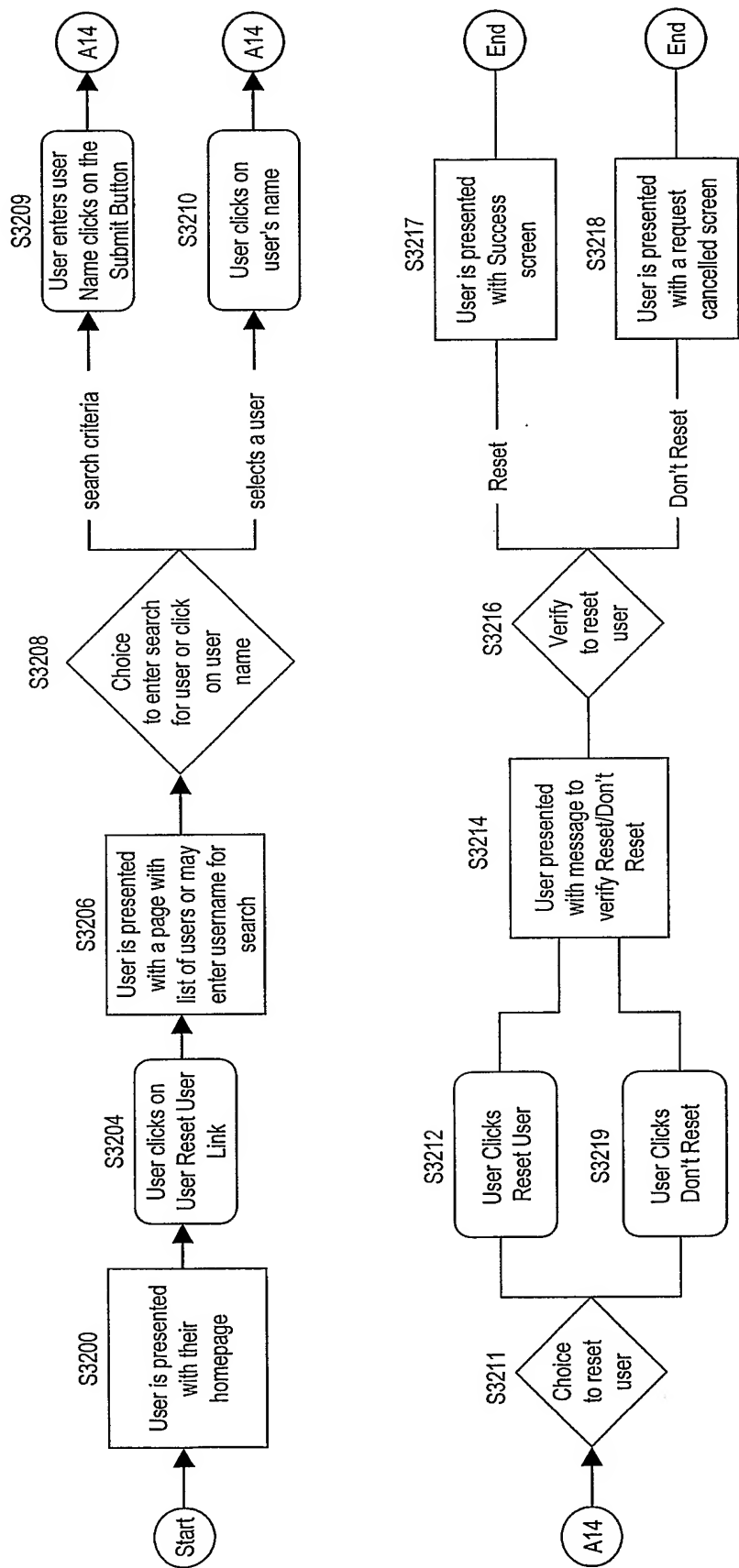


Fig. 21 Reset User



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 22A Transfer a PIC

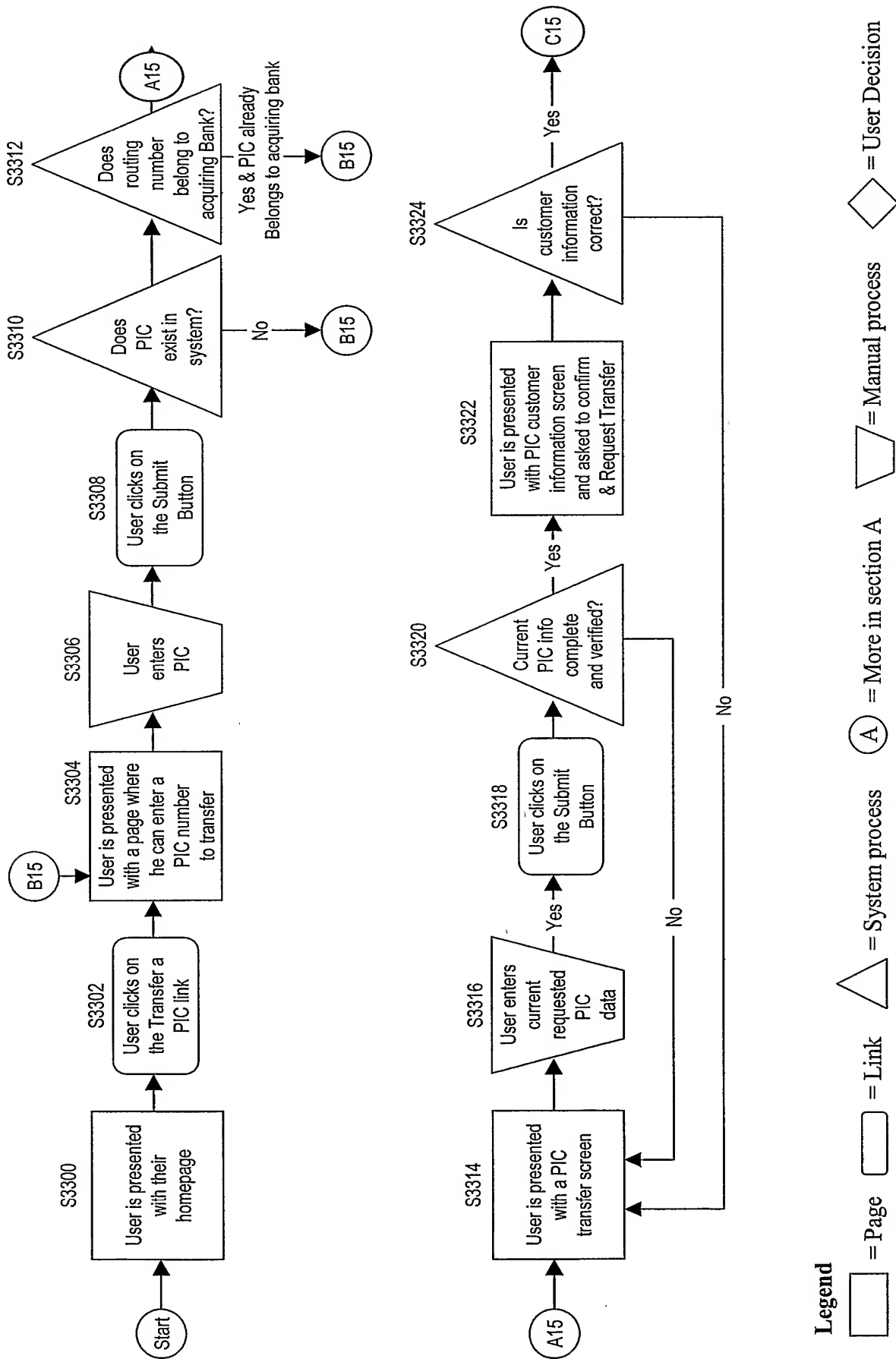


Fig. 22B Transfer a PIC

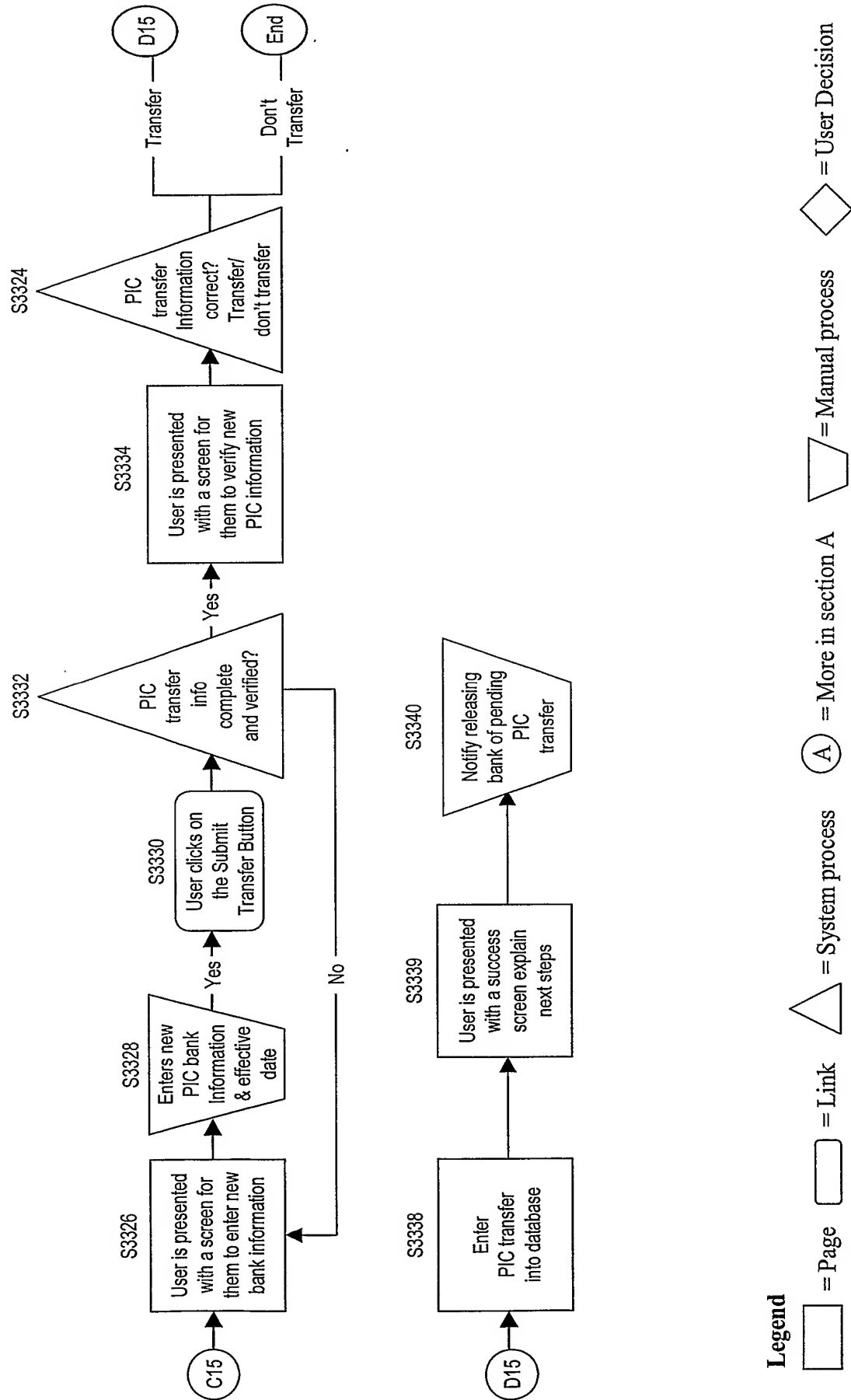


Fig. 23 Create A PIC

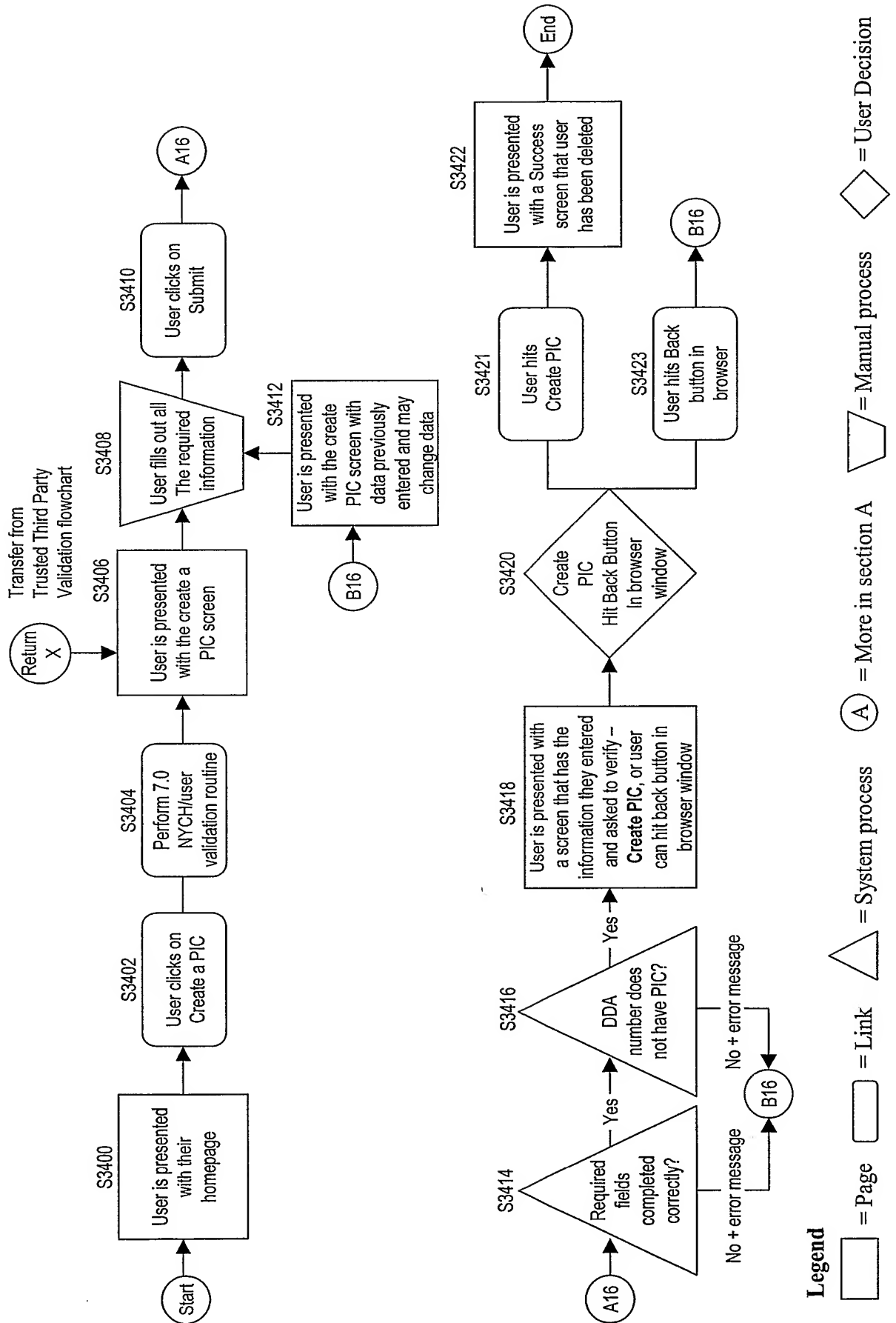
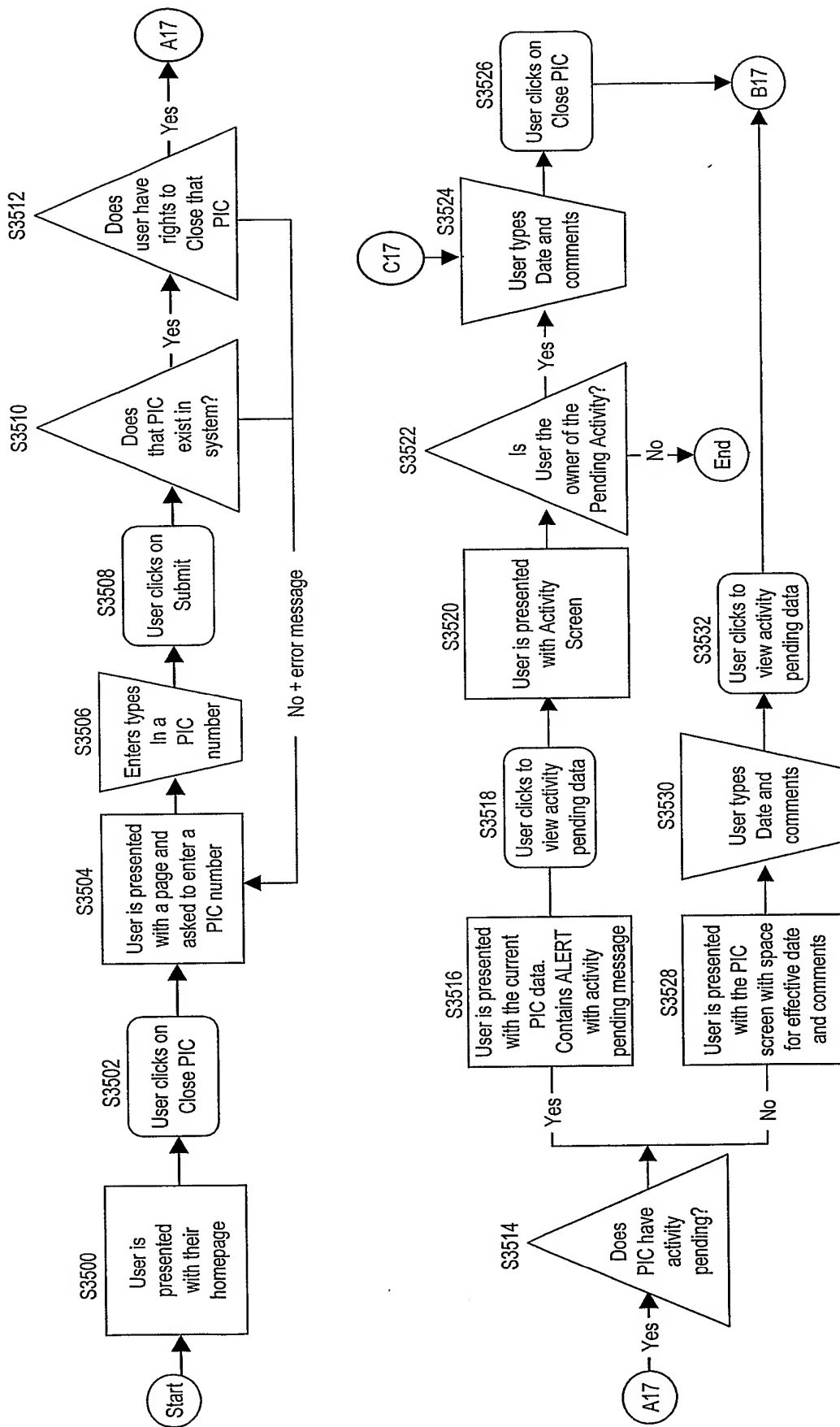


Fig. 24A Close A PIC



Legend



= Page



= Link



= System process



= More in section A

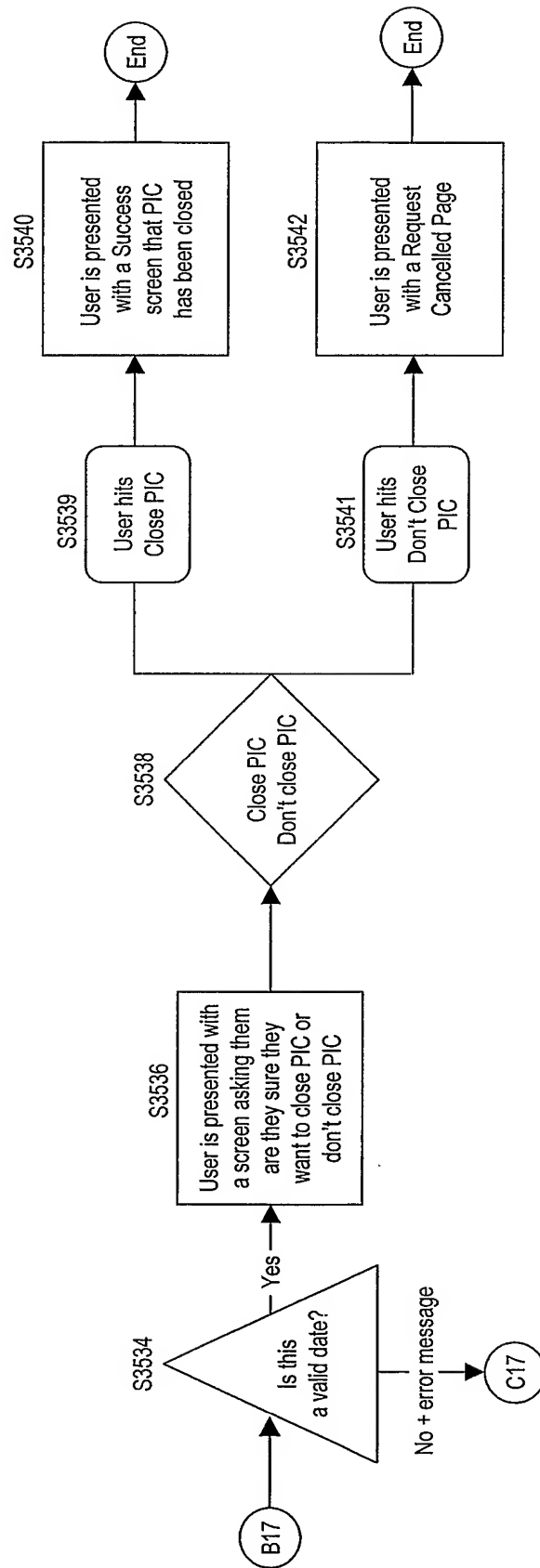


= Manual process



= User Decision

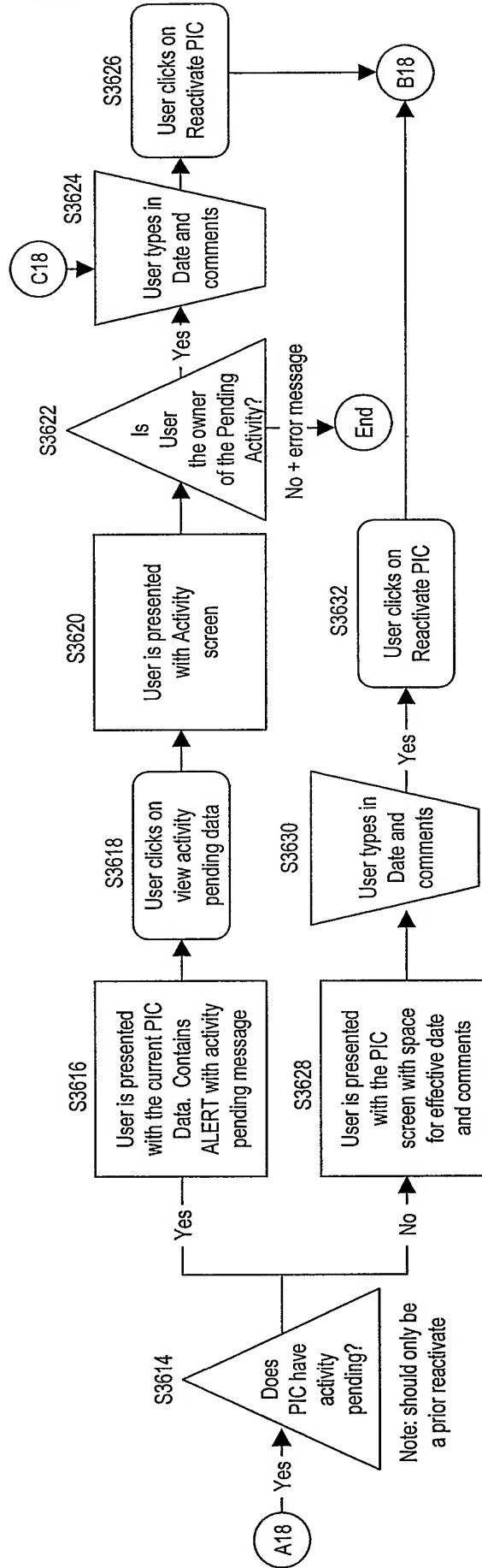
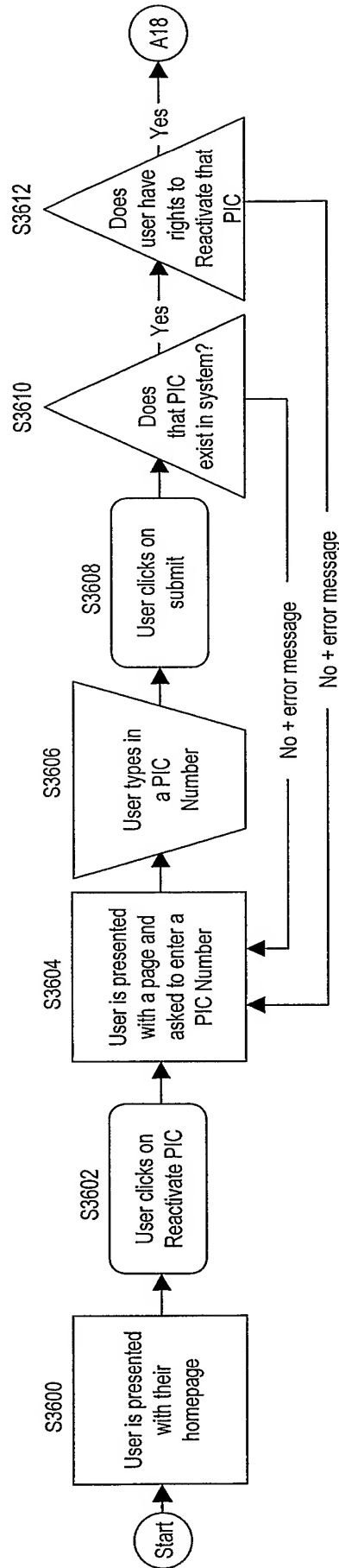
Fig. 24B Close A PIC



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

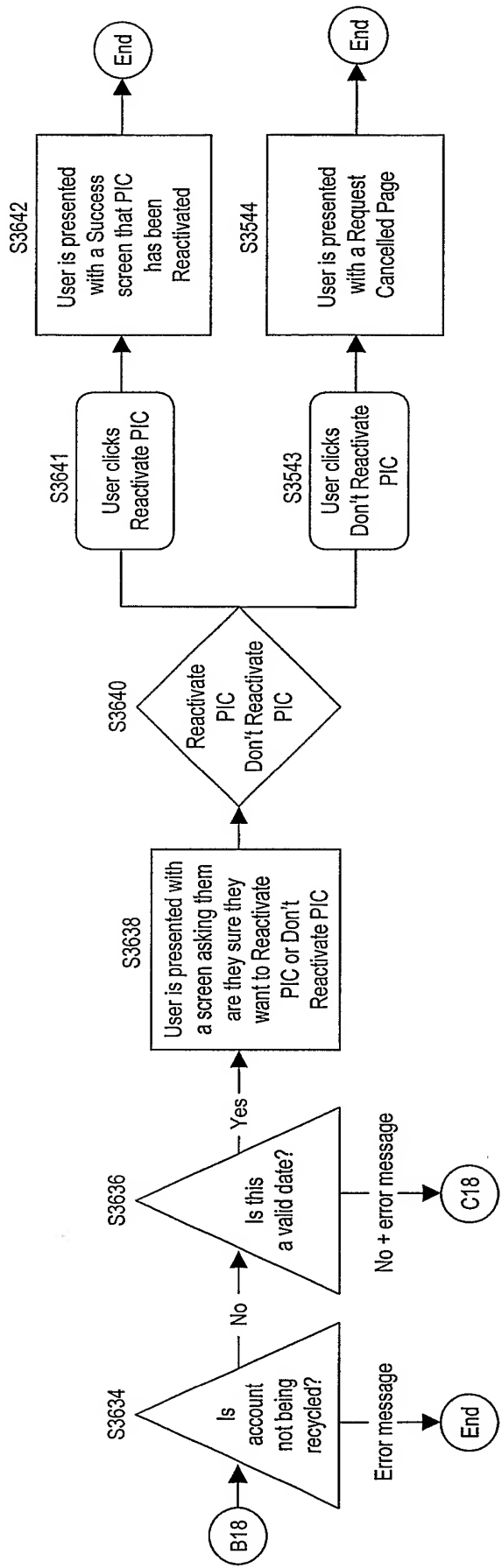
Fig. 25A Reactivate A PIC



Legend



Fig. 25B Reactivate A PIC



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 26 Contest/Release A PIC Pending Transfers

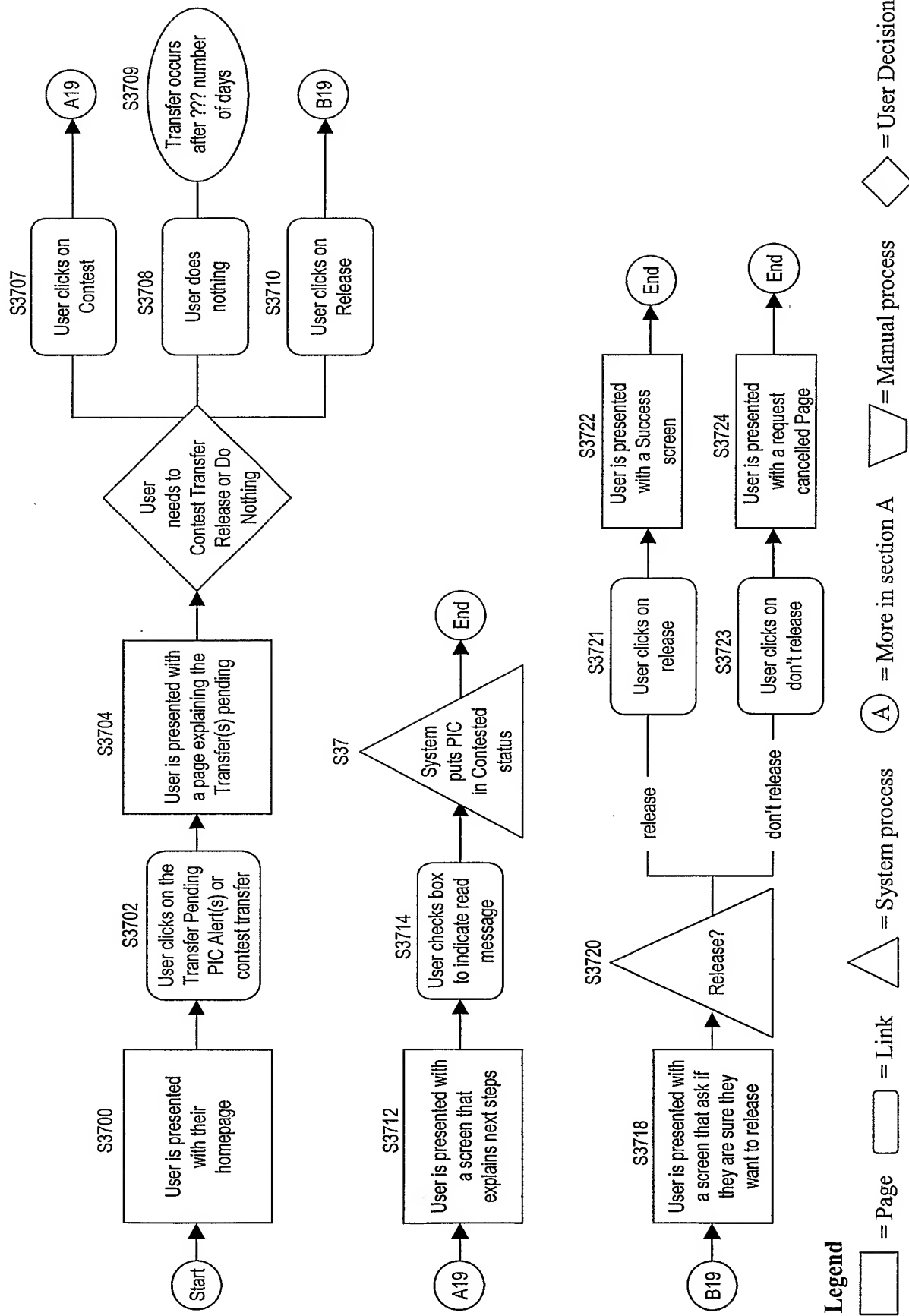
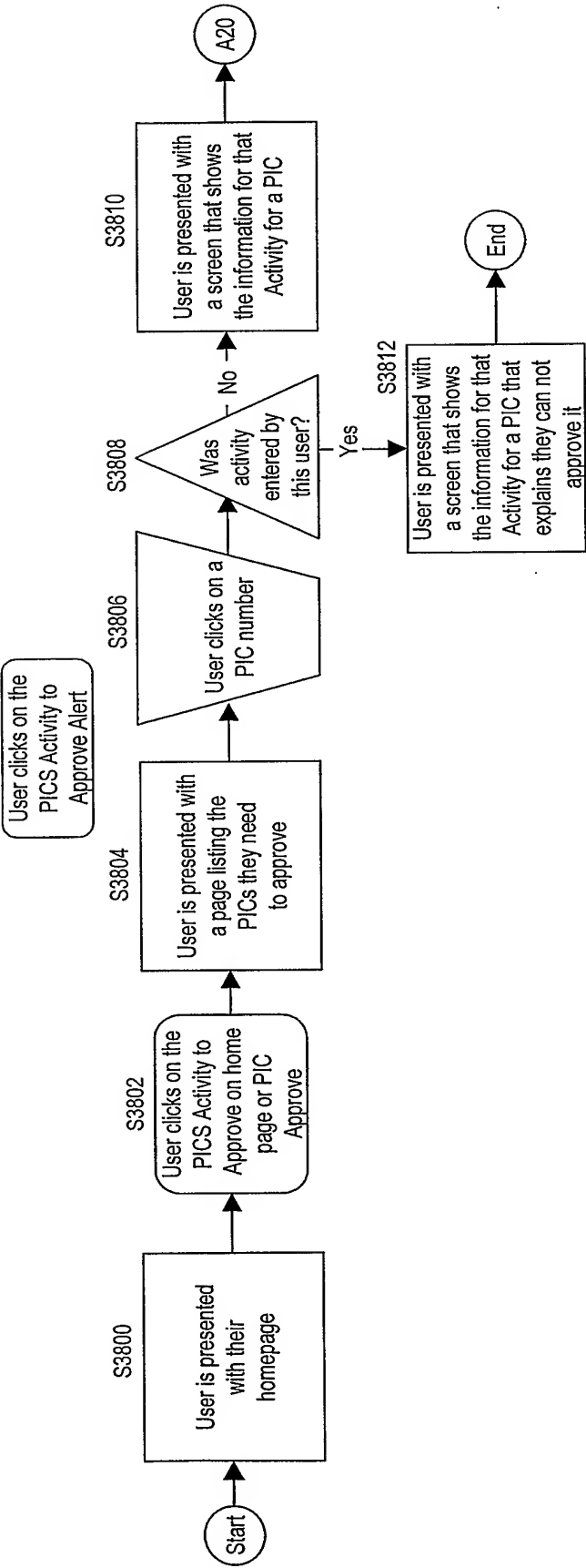


Fig. 27A Approve A PIC Activity



Legend



= Page

= Link



= System process



= More in section A



= Manual process



= User Decision

Fig. 27B Approve A PIC Activity

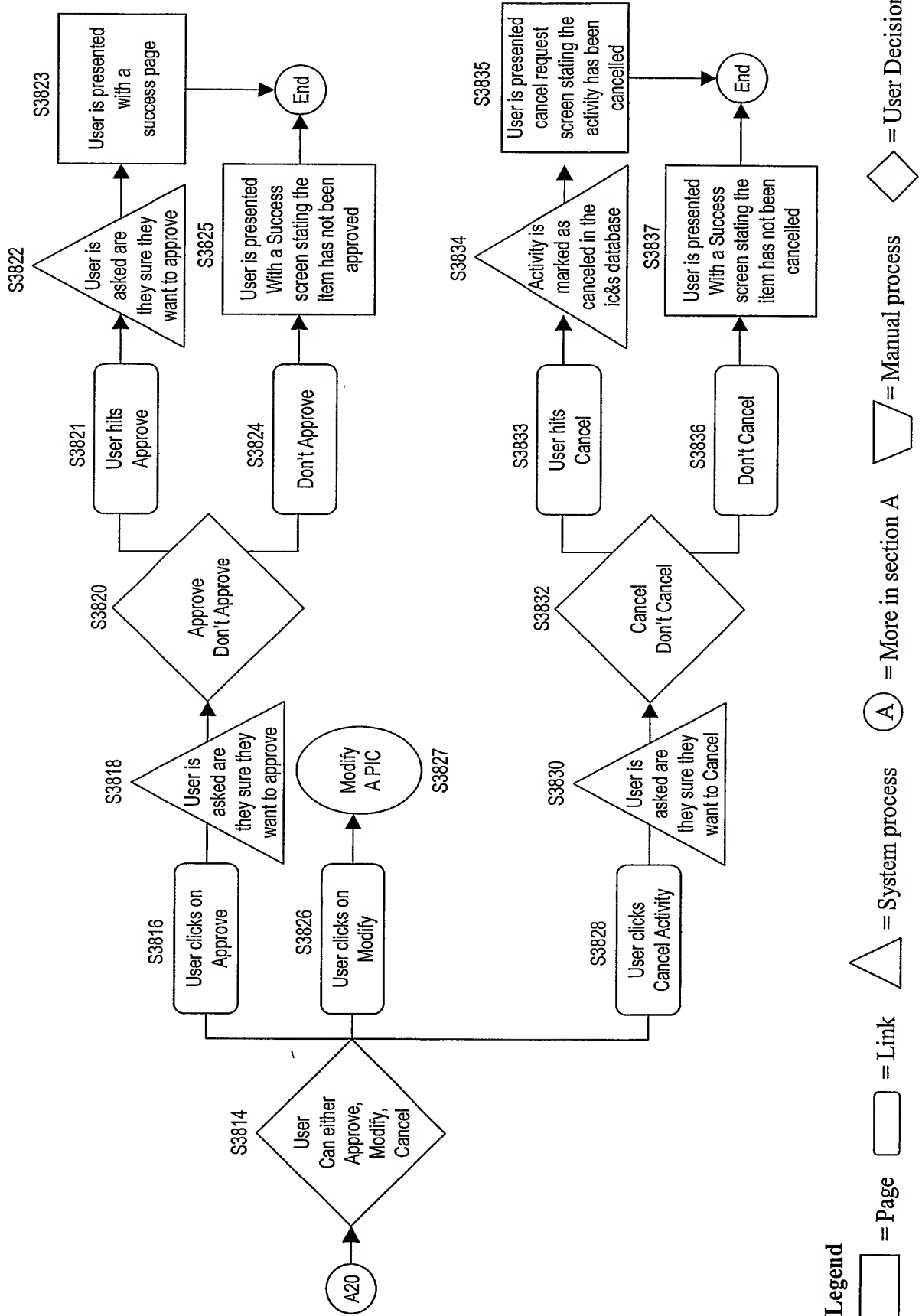


Fig. 28A Find A PIC

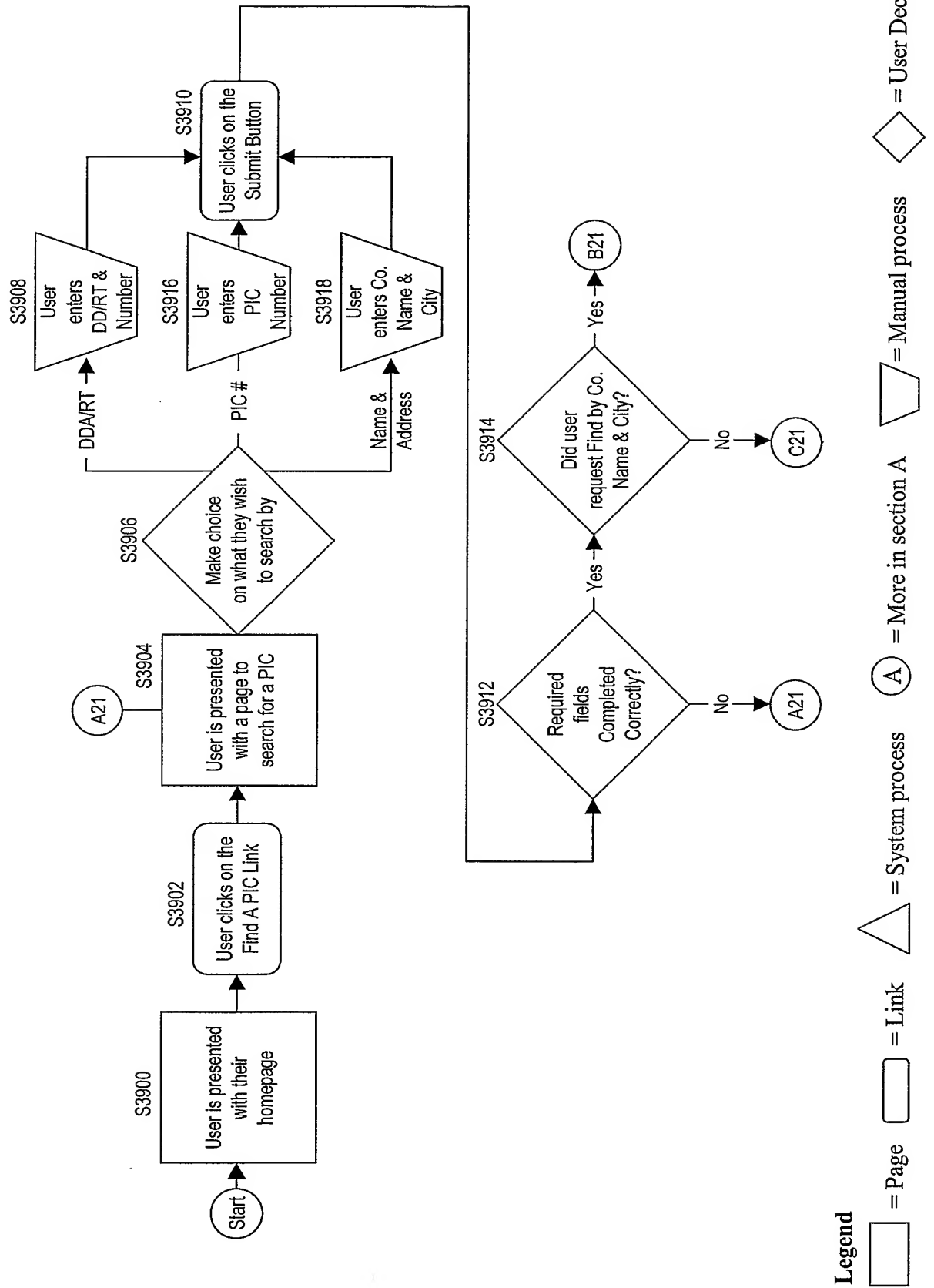
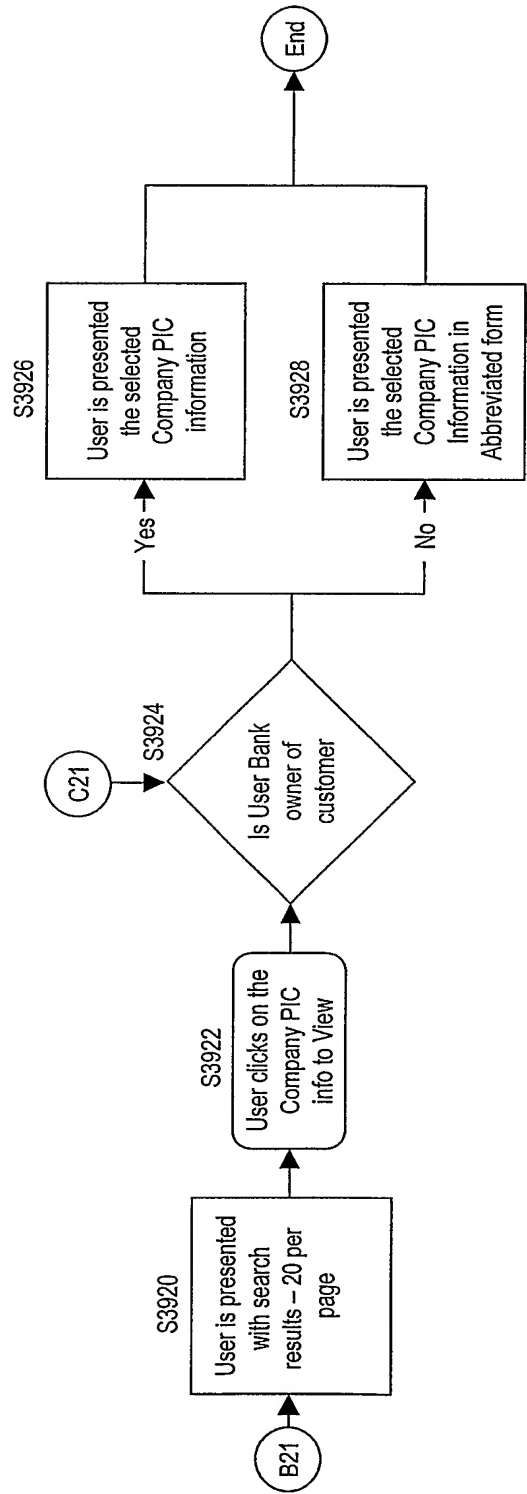


Fig. 28B Find A PIC



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 29 View User Profile Activity Log

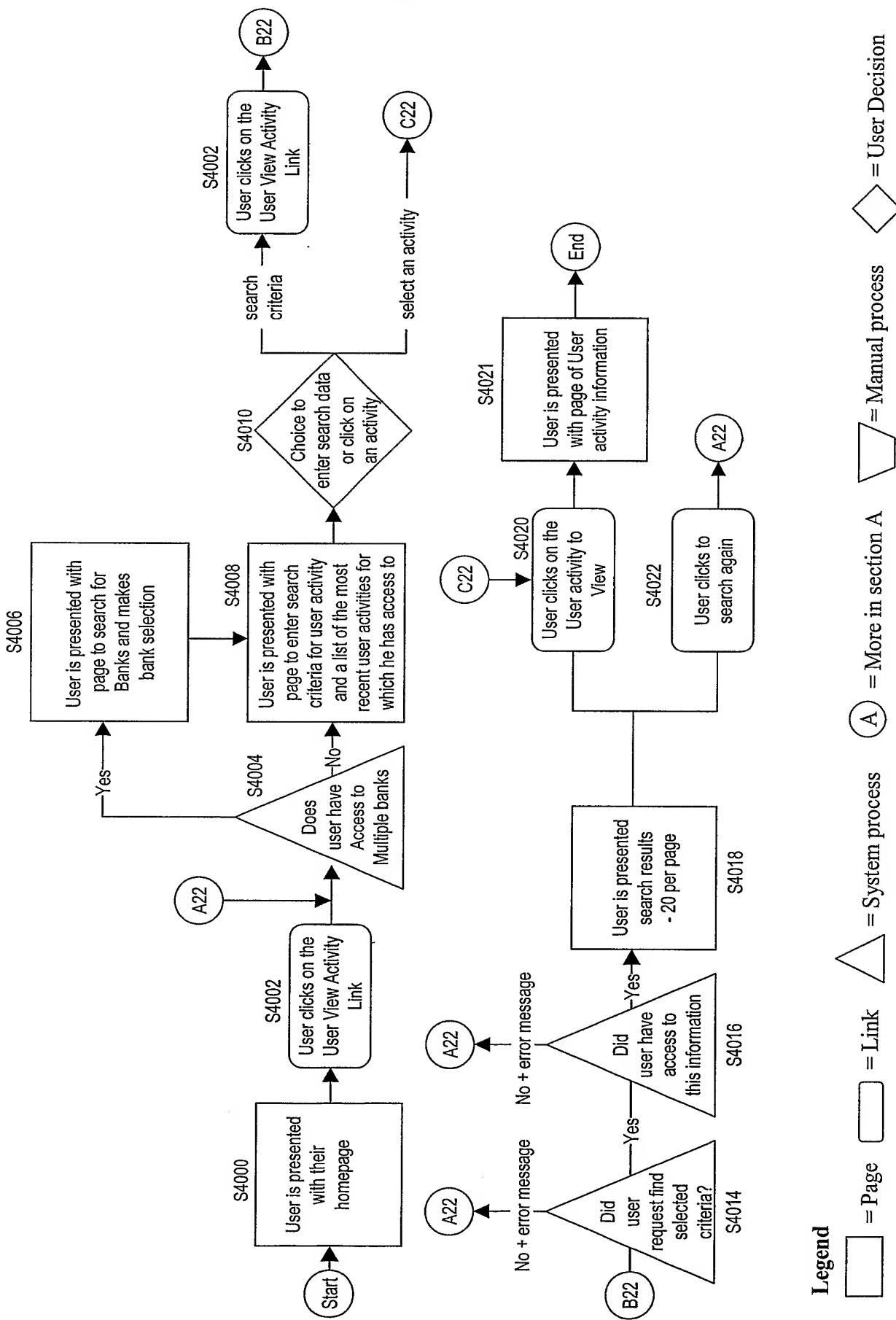


Fig. 30A Modify A PIC

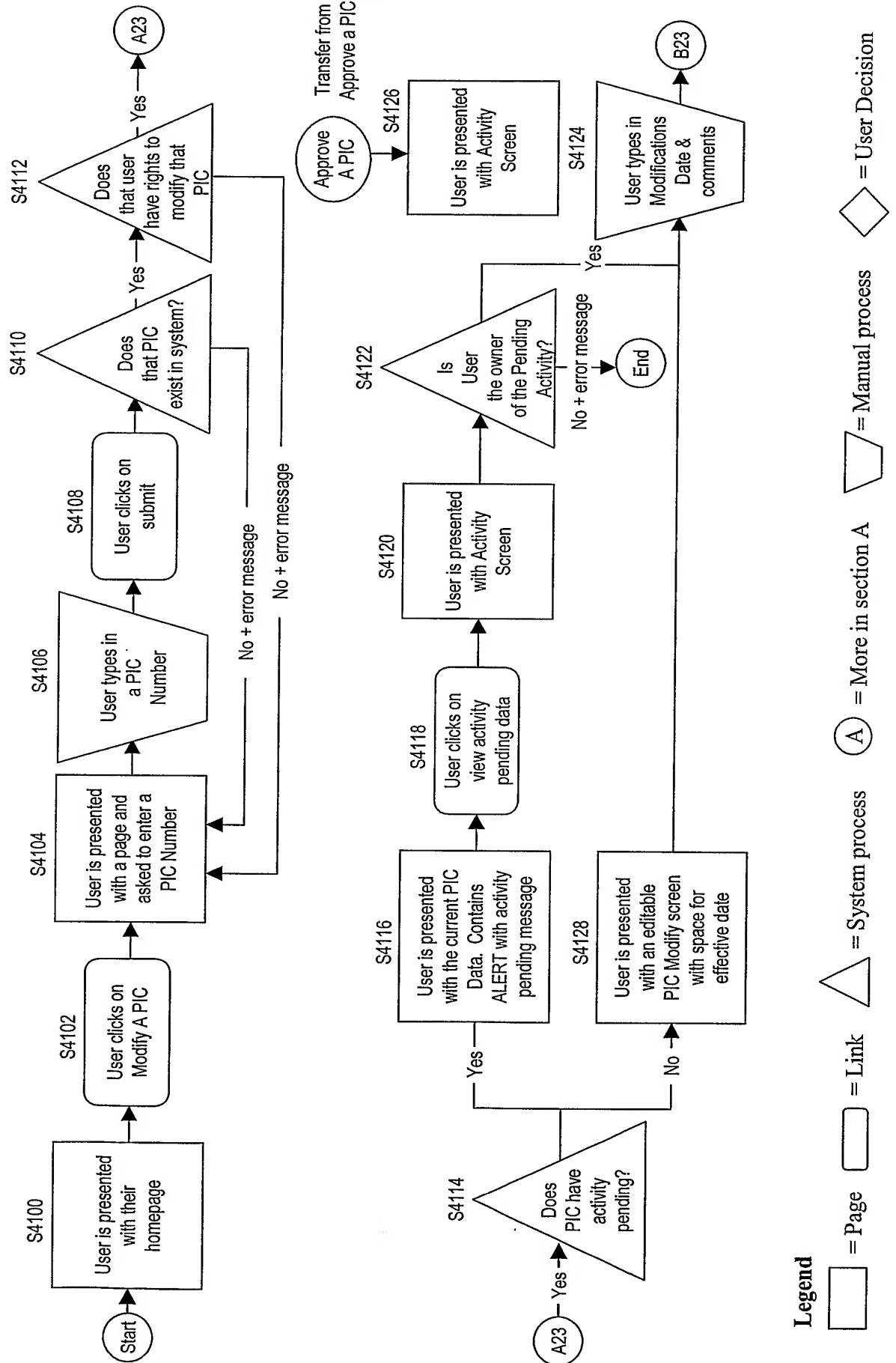
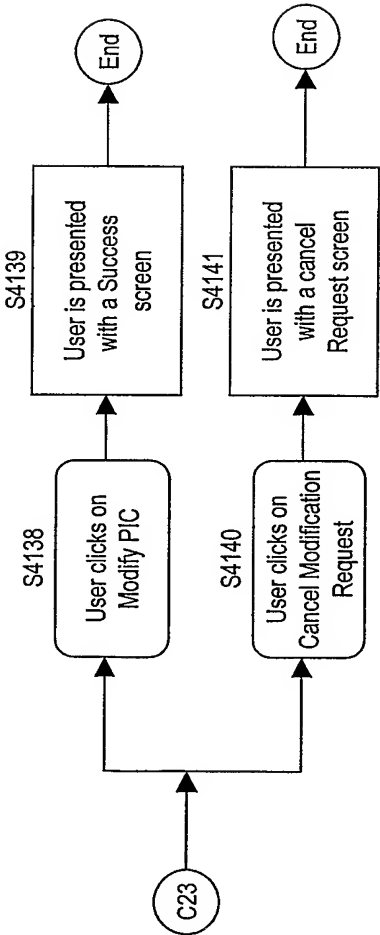
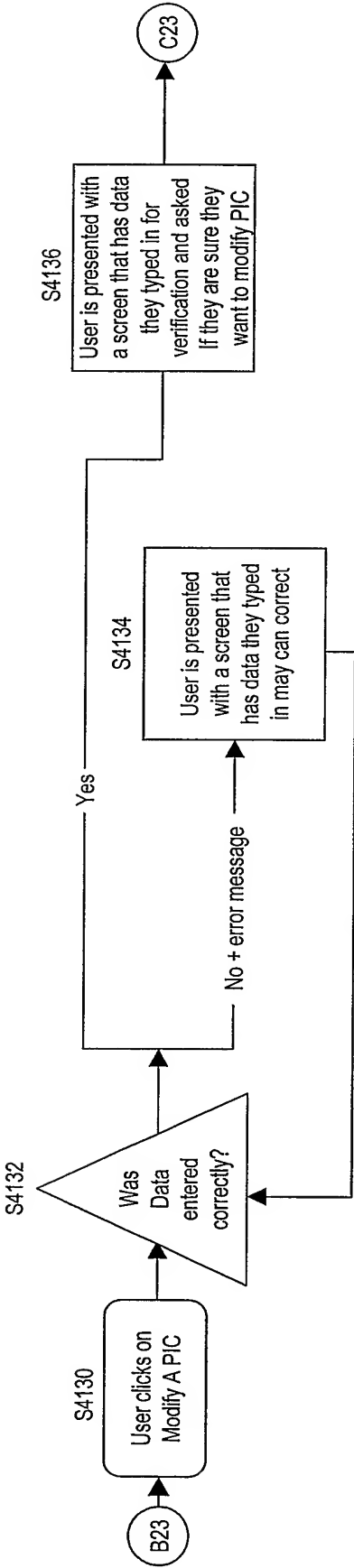


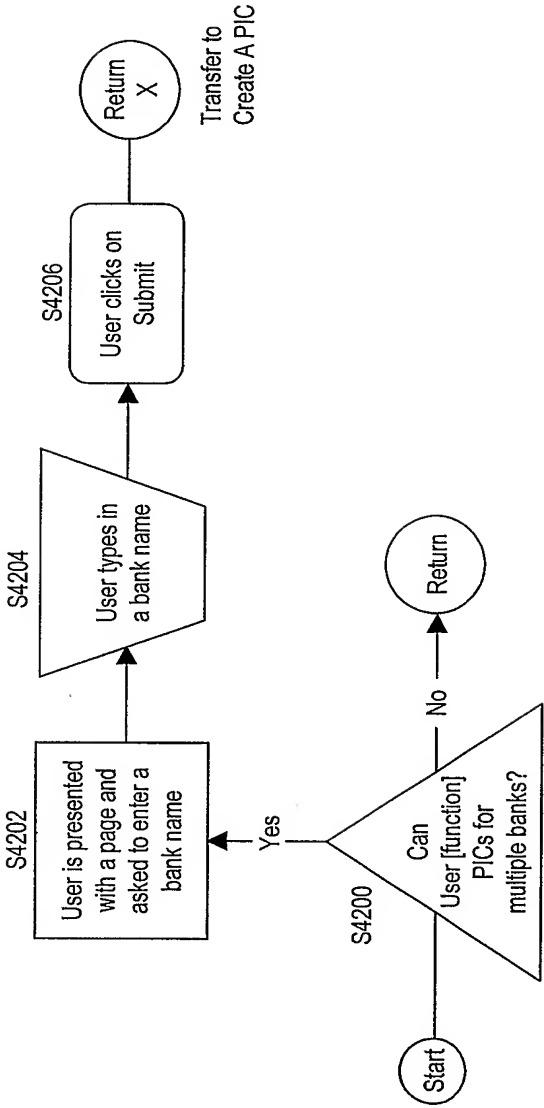
Fig. 30B Modify A PIC



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

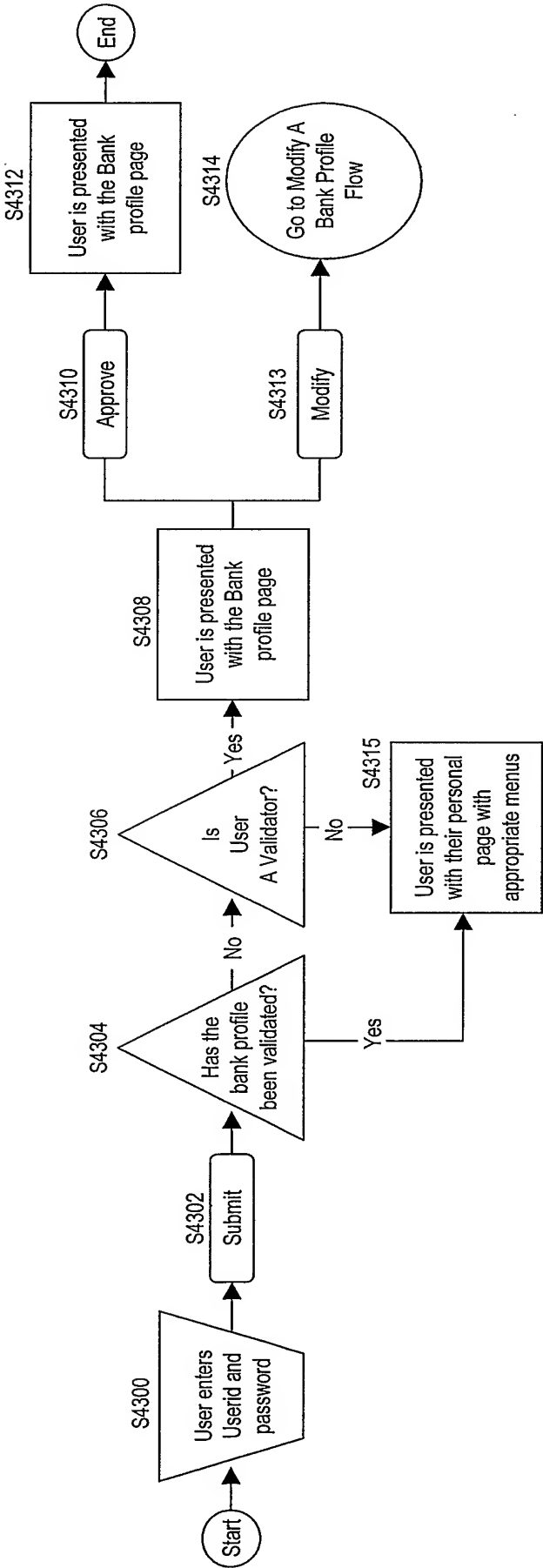
Fig. 31 Trusted Third Party



Legend

- = Page = Link = System process = More in section A = Manual process = User Decision

Fig. 32 Validate Bank Profile



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 33A Approve A Bank Profile Modification

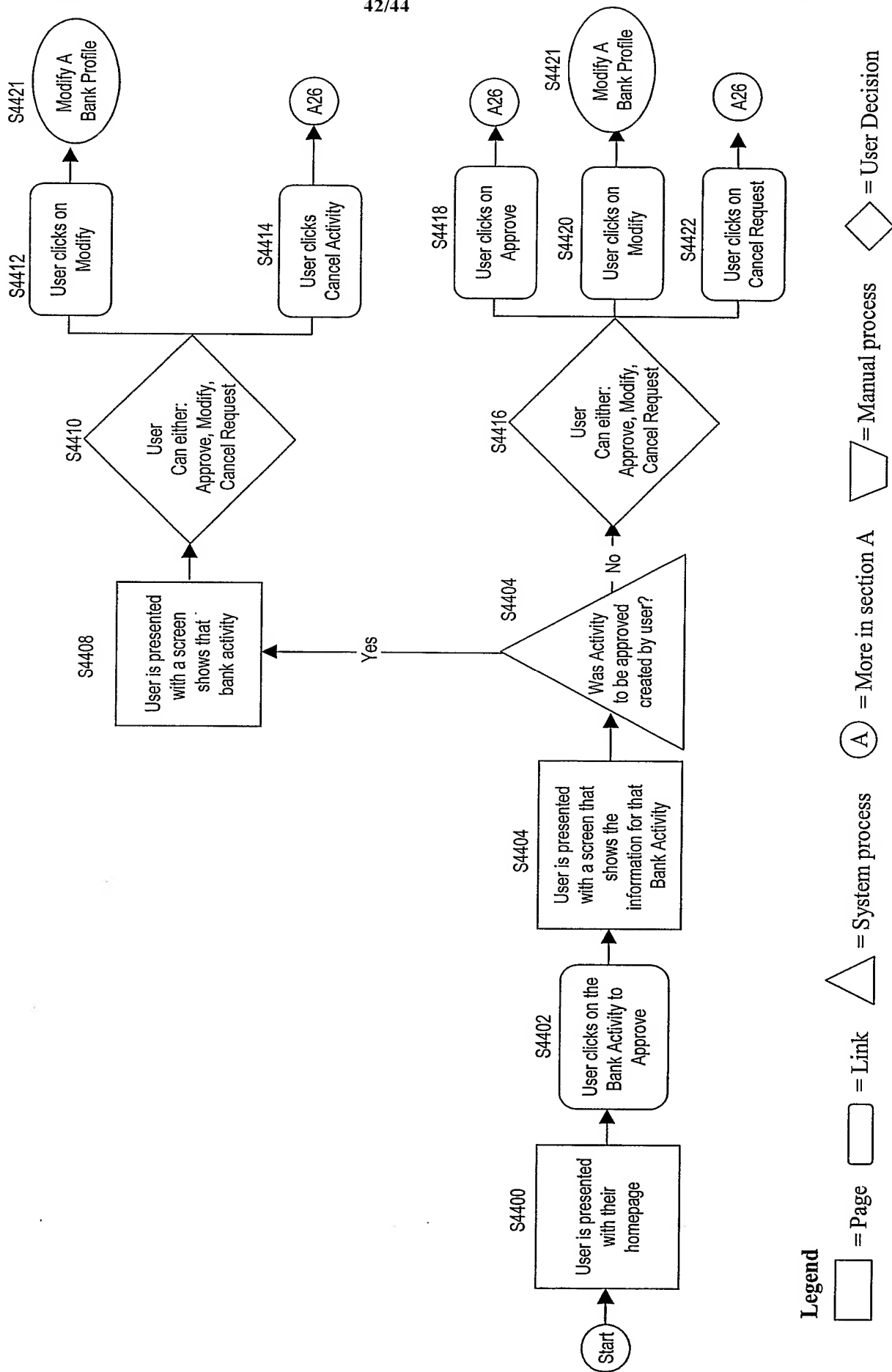
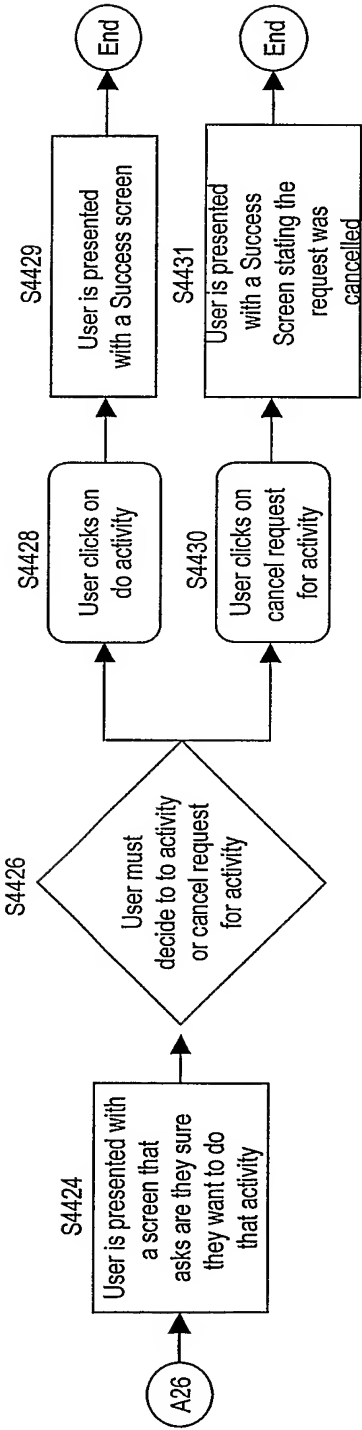


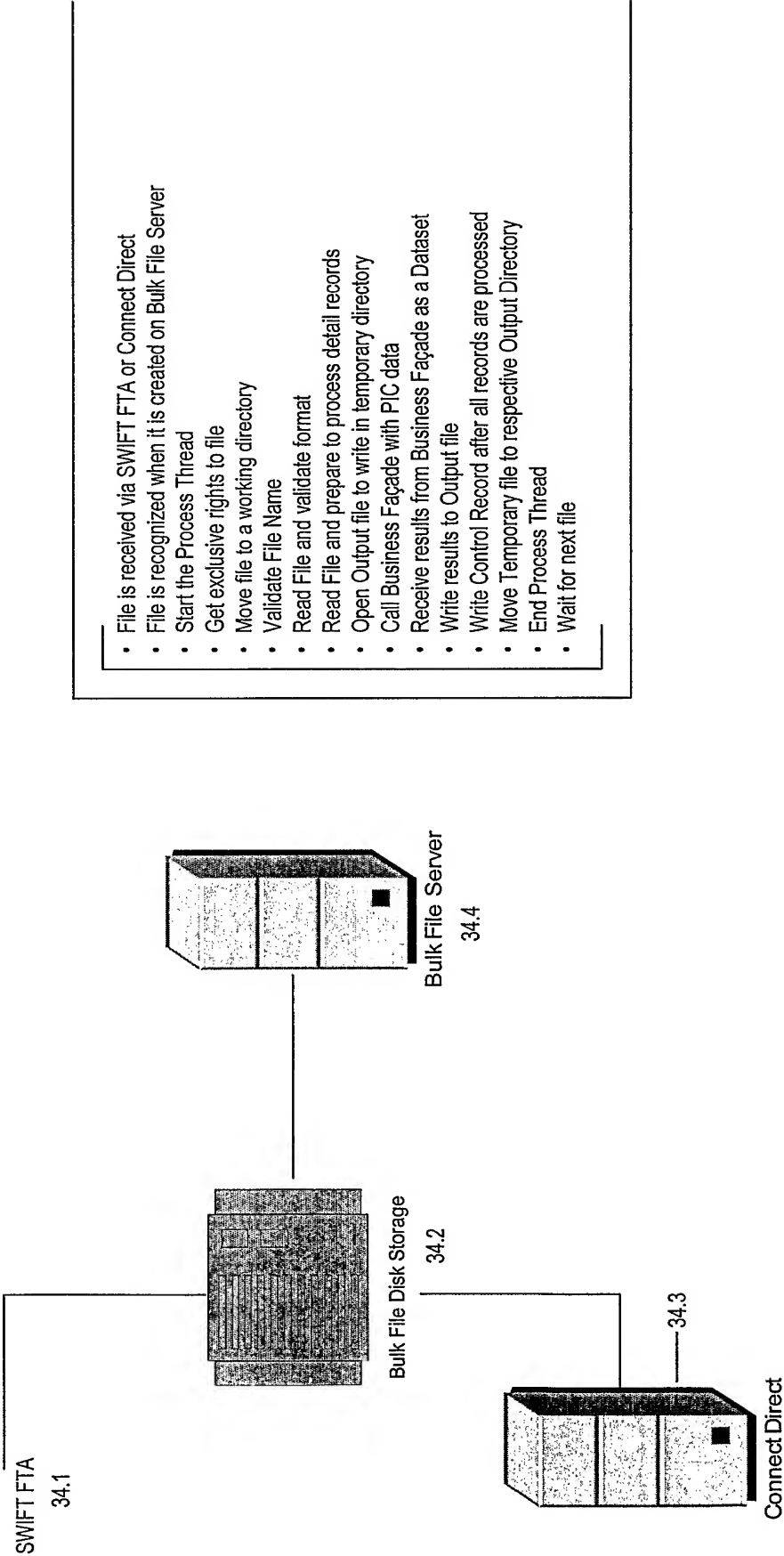
Fig. 33B Approve A Bank Profile Modification



Legend

- = Page
- = Link
- = System process
- = More in section A
- = Manual process
- = User Decision

Fig. 34 PIC BATCH SERVICE



(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
6 November 2003 (06.11.2003)

PCT

(10) International Publication Number
WO 2003/091849 A3

(51) International Patent Classification⁷: **G06F 17/60**

07950 (US). **FRIEDMAN, Kenneth, H.**; 22 Exeter Drive, Marlboro, NJ 07746 (US). **PALMESE, Paul**; 500 Pierce Street, Oceanside NY 11572 (US).

(21) International Application Number:
PCT/US2003/012832

(22) International Filing Date: 23 April 2003 (23.04.2003)

(74) Agents: **CLAYTON, Ronald, A.** et al.; Fitzpatrick, Cella, Harper & Scinto, 30 Rockefeller Plaza, New York, NY 10112-3801 (US).

(25) Filing Language: English

(26) Publication Language: English

(84) Designated States (*regional*): European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR).

(30) Priority Data:
60/374,936 23 April 2002 (23.04.2002) US

Published:
— with international search report

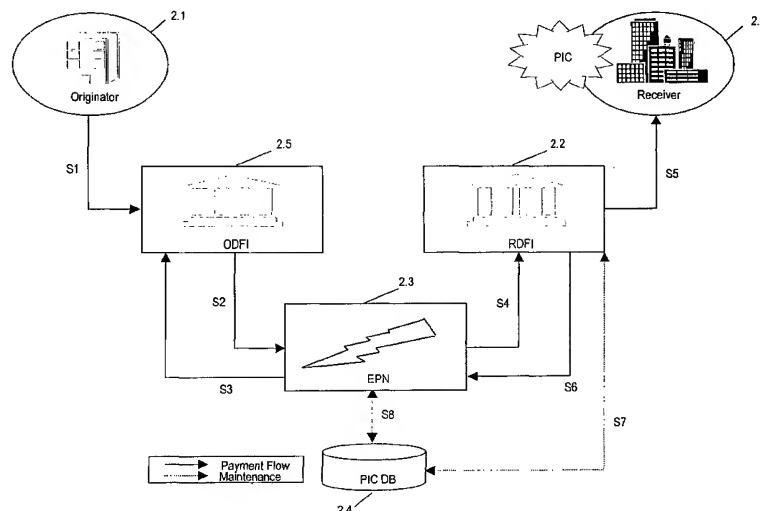
(71) Applicant: **THE CLEARING HOUSE SERVICE COMPANY L.L.C.** [US/US]; 100 Broad Street, New York, NY 10004 (US).

(88) Date of publication of the international search report:
5 February 2004

(72) Inventors: **THOMAS, George, F.**; 111 Middlesex Avenue, Oakdale, NY 11769 (US). **WOOD, Albert, G.**; 8 Landview Drive, Kings Park, NY 11754 (US). **PAWEL-CZYK, Joseph, S.**; 9 Wyckoff Way, Morris Plains, NJ

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: PAYMENT IDENTIFICATION CODE SYSTEM



(57) Abstract: A method for effecting electronic payment, safeguarding banking and account information, while utilizing existing payment systems. The method comprises generating a system routing number and a payment identification code (PIC) (2.3) relating to the beneficiary's account information, distributing payment identification codes to the existing payment system and financial institutions (2.1) owning the account related to the payment identification codes, and the originator (2.1) receiving a system routing number and the beneficiary's PIC number (2.1). The method also includes the originator (2.1) communicating a payment instruction to a financial institution of the originator (2.1), wherein the payment instruction includes the system routing number beneficiary's payment identification code, the originator's financial institution receiving the payment instruction from the originator (2.1).

WO 2003/091849 A3

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US03/12832

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 17/60

US CL : 705/1, 24, 26, 30, 34, 35, 39, 40, 42, and 44.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/1, 24, 26, 30, 34, 35, 39, 40, 42, and 44.

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Please See Continuation Sheet

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|------------|--|-----------------------|
| X | US 5,999,625 A (BELLARE et al) 07 December 1999 (07.12.1999) col. 1, line 15-col. 2, line 27, col. 3, lines 49-63, col. 5, lines 18-30, col. 6, lines 24-33, and col. 9, lines 53, col. 3, line 54-col. 10, line 56, col. 21, line 26-col. col. 23, line 6, and col. 27, line 47-col. 30, line 63. | 1-15 |
| X | US 5,920,847 A (KOLLING et al) 06 July 1999 (06.07.1999, col. 1, line 15-col. 2, line 53, col. 3, line 54-col. 10, line 56, col. 21, line 26-col. col. 23, line 6, and col. 27, line 47-col. 30, line 63. | 1-15 |
| X | US 6,138,107 A (ELGAMAL) 24 October 2000 (24.10.2000) see entire reference. | 1-15 |
| X | US 2001/0024157 A1 (HANSMANN et al) 27 September 2001 (27. 09.2001) see entire reference. | 1-15 |



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"B" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T"

later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X"

document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y"

document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&"

document member of the same patent family

Date of the actual completion of the international search

21 September 2003 (21.09.2003)

Date of mailing of the international search report

19 NOV 2003

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Facsimile No. (703)305-3230

Authorized officer

Ella Colbert

Telephone No. 703-308-7064

INTERNATIONAL SEARCH REPORT

PCT/US03/12832

Continuation of B. FIELDS SEARCHED Item 3:

WEST

search terms: routing number, electronic bill payment, secure, bank, institution, financial, payee, beneficiary, pic, payment identification code, id, account information, list, valid, copy, EPN, and database.